

Trabajo de fin de grado

Plataforma web para aprender a programar videojuegos

Grado en ingeniería Informática
Sistemas de información

Autor: Gerard Belenguer Prous

Tutor: Marc Alier Forment

19/06/18

Índice

| | | |
|--------|---|----|
| 1. | Introducción..... | 3 |
| 1.1. | Contexto..... | 3 |
| 1.2. | Actores implicados (stakeholders)..... | 3 |
| 2. | Formulación del problema..... | 4 |
| 2.1. | Problema..... | 4 |
| 3. | Estado del arte..... | 4 |
| 3.1. | Inicios..... | 4 |
| 3.2. | Estado actual..... | 4 |
| 3.3. | Estudio de mercado..... | 5 |
| 4. | Alcance del proyecto..... | 10 |
| 4.1. | Meta final..... | 10 |
| 4.2. | Obstáculos y riesgos..... | 10 |
| 5. | Metodología..... | 11 |
| 5.1. | Scrum modificado..... | 11 |
| 5.2. | Métodos de validación..... | 13 |
| 6. | Planificación Temporal..... | 13 |
| 6.1. | Tareas..... | 13 |
| 6.2. | Diagrama de Gantt..... | 16 |
| 6.2.1. | Dependencias..... | 17 |
| 7. | Recursos..... | 17 |
| 7.1. | Recursos Humanos..... | 17 |
| 7.2. | Hardware..... | 17 |
| 7.3. | Software..... | 18 |
| 8. | Valoración de alternativas y plan de acción..... | 18 |
| 8.1. | Falta de tiempo..... | 18 |
| 8.2. | Problemas con la tecnología..... | 19 |
| 8.3. | Fracaso de la plataforma..... | 19 |
| 8.4. | Otros..... | 19 |
| 9. | Presupuesto y sostenibilidad..... | 20 |
| 9.1. | Autoevaluación competencia de sostenibilidad..... | 20 |
| 9.2. | Matriz de sostenibilidad..... | 21 |
| 9.2.1. | Impacto Ambiental..... | 22 |
| 9.2.2. | Impacto Económico..... | 23 |
| 9.2.3. | Impacto Social..... | 26 |
| 10. | Seguimiento..... | 27 |
| 10.1. | Desviaciones..... | 27 |
| 10.2. | Impacto..... | 29 |
| 11. | Implementación de la página..... | 30 |

| | | |
|-------|--|----|
| 11.1. | Preparación del dominio y servidor..... | 30 |
| 11.2. | Aprender Wordpress..... | 32 |
| 11.3. | Diseño de la página..... | 33 |
| 11.4. | Implementación de la base de la web..... | 34 |
| 11.5. | Plugins..... | 37 |
| 11.6. | Problemas a resolver..... | 42 |
| 12. | Creación de contenidos didácticos..... | 45 |
| 12.1. | Objetivos..... | 45 |
| 12.2. | Proceso..... | 45 |
| 13. | Gestión del sistema de información y presencia online..... | 46 |
| 13.1. | SEO..... | 47 |
| 13.2. | Estadísticas de la plataforma y toma de decisiones..... | 52 |
| 13.3. | Decisiones de monetización..... | 57 |
| 13.4. | Gestión de las listas de correo..... | 58 |
| 13.5. | Generación de leads y embudo de conversión..... | 61 |
| 13.6. | Gestión de las redes sociales..... | 62 |
| 14. | Conclusiones..... | 65 |
| 14.1. | Cumplimiento de los objetivos..... | 66 |
| 14.2. | Conocimientos aplicados y adquiridos..... | 66 |
| 14.3. | El futuro de Aprender Unity..... | 67 |
| 15. | Referencias..... | 68 |
| 15.1. | Lista de figuras..... | 69 |

1.Introducción

El objetivo de este proyecto es el de crear una plataforma web donde programadores de videojuegos puedan acudir para expandir sus conocimientos, aclarar sus dudas y conocer a otros desarrolladores.

1.1.Contexto

El desarrollo de videojuegos se está democratizando. Cada vez existen más herramientas gratuitas y fáciles de usar que están provocando un gran aumento de gente interesada en este ámbito.

En muchos casos los desarrolladores novatos no tienen un amplio conocimiento sobre programación y además tampoco saben utilizar las herramientas que tienen al alcance. Y aunque existen muchas fuentes de conocimiento e información hay pocas de calidad, sobretodo en español.

1.2.Actores implicados

Durante las distintas fases del desarrollo de este proyecto habrán diversos actores implicados (o stakeholders¹). Estos serán los siguientes:

- 1) **Director del proyecto:** El director del proyecto es Marc Alier Forment. Será él quien guíe, supervise y asesore este proyecto en todas sus fases con el objetivo de ayudar a garantizar que se consigan cumplir todos los objetivos de la mejor forma posible.
- 2) **Desarrollador:** Es el encargado de desarrollar la plataforma web, tomar las decisiones necesarias para llevar el proyecto adelante (por supuesto con la ayuda del director) y de proporcionar los contenidos docentes de la plataforma.
- 3) **Usuarios finales:** Estos serán todos aquellos desarrolladores que acudirán a la plataforma ya sea para aprender, para resolver alguna duda o para contactar con otros desarrolladores. El producto está dirigido a ellos, serán ellos quien usen y se beneficien de la plataforma.

¹ stakeholders: Término inglés popularizado por R.E. Freeman, que lo usó para referirse a “quienes son afectados o pueden ser afectados por las actividades de una empresa”

2. Formulación del problema

2.1. Problema

Como hemos visto en el contexto, el desarrollo de videojuegos está a la alza gracias a lo accesible que se está volviendo. Y cada día más y más gente se propone aprender como hacerlo para así poder crear ese juego que han estado imaginando desde que eran pequeños.

El problema que esto plantea es que aunque existen muchas fuentes de información esta no suele estar bien presentada. Y aunque lo esté nunca está centralizada, es decir, no hay un sitio que realmente unifique todos los conocimientos que uno necesita para aventurarse en el mundo de la programación de videojuegos. Esto también dificulta la posibilidad de poder contactar y conocer otros desarrolladores.

Esto por no hablar de la barrera del idioma. Obviamente este ámbito está puramente dominado por el inglés así que la gente que no tiene un buen nivel del idioma está muy limitada en cuanto a opciones de aprendizaje.

3. Estado del arte

3.1. Inicios

Cuando los videojuegos empezaron a popularizarse a principios de los años 80 sus programadores eran verdaderos expertos en todos los campos de la ingeniería informática. Tenían que ser verdaderos expertos en hardware para así poder entender cómo funcionaba la máquina sobre la que desarrollaban y expertos programadores para poder construir un juego usando el conocimiento que tenían sobre el hardware. Además las herramientas eran muy limitadas, no había mucha información al respecto, para colmo poca gente disponía de equipos para poder programar y además internet no había llegado a las casas del gran público.

Por ese motivo al principio poca gente era capaz de desarrollar videojuegos, pero con suerte con el tiempo ha ido mejorando hasta llegar al punto donde para desarrollar videojuegos no es ni siquiera necesario conocer el hardware sobre el que vamos a ejecutar nuestro juego.

3.2. Estado actual

Ya hace años que cualquiera que se lo proponga puede aprender a programar videojuegos pero a día de hoy es más fácil que nunca. Hoy en día contamos con un montón de motores de videojuegos gratuitos, de fácil uso y ya más que consolidados en la industria tales como: Unity, Unreal Engine, Game Maker Studio, Cryengine...

Estos motores simplifican todos los ámbitos del desarrollo y ponen al alcance de todos herramientas antes sólo accesibles para verdaderos expertos en la materia. En las referencias hay un artículo que habla de la democratización del desarrollo de videojuegos [1].

En consecuencia se ha disparado el número de gente interesada en la programación de videojuegos y con ello la oferta y demanda de recursos didácticos para el desarrollo de videojuegos. Una oferta que con puntuales excepciones es de baja calidad, no está centralizada en un solo sitio web y que cuenta con malos métodos didácticos que no ayudan al aprendizaje.

3.3. Estudio de mercado

La cantidad de herramientas, motores de videojuegos y posibilidades que existen hoy en día es muy grande. Así que este estudio de mercado se va a centrar en Unity, el motor sobre el que se va a basar la plataforma. Ya que es el más usado y en la que más experiencia tengo (llevo trabajando un año en una pequeña empresa de videojuegos usando este motor). Además nos centraremos sobretodo en los recursos docentes en español ya que en principio la plataforma sólo estará en español.

Así que: cuáles son las principales formas de aprender por internet a usar y programar en Unity?

- 1) **Tutoriales en Youtube:** A día de hoy es la fuente principal de información. En Youtube se hacen tutoriales en forma de vídeo normalmente con una narración en forma de audio de lo que se va haciendo. Esta es la forma más visual de aprender y la que menos esfuerzo requiere por parte del espectador ya que en vez de leer documentos y estudiar código comentado es capaz de ver como alguien les narra y explica lo que quieren aprender.

Hay que remarcar que los canales de Youtube realmente grandes y con contenido de mayor calidad (mejor producción, más estructurados, más acertados y mejor narrados) son de habla inglesa. El mayor referente a día de hoy es:

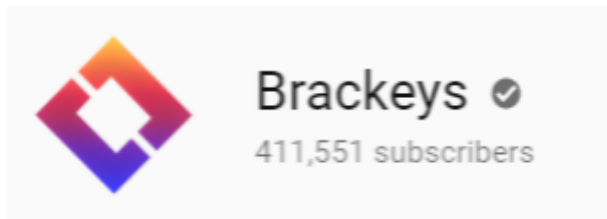


Figura 1. Brackeys 411551 suscriptores

El cual crea un contenido de mucha calidad pero a la vez de una complejidad muy baja.

Luego le siguen:

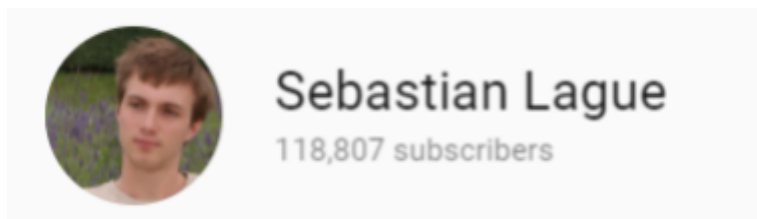


Figura 2. Sebastian Lague 118807 suscriptores

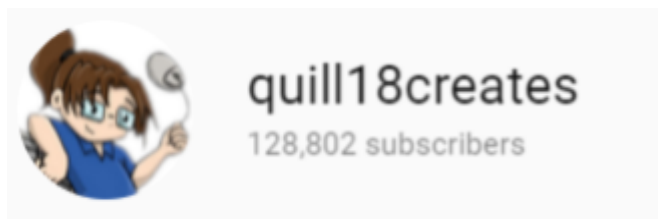


Figura 3. Quill18creates 128802 suscriptores

Estos creadores a diferencia de Brackeys (Figura 1) producen contenido de una complejidad muchísimo más alta y seguramente por eso no tengan tantos suscriptores/seguidores. Parece que los grandes números se consiguen con material que el gran número de gente que se está adentrando en el desarrollo de videojuegos pueda entender.

Ahora que hemos visto la clase de números que tienen los canales con más seguidores de lengua inglesa y que tipo de contenido producen vamos a echar un vistazo a los de habla hispana que son los que realmente nos interesan. En general los canales españoles tienen una calidad mucho peor y a la vez muchos menos seguidores. Obviamente hay mucho más público potencial si el canal es de habla inglesa. Pero aún así, si consideramos que el contenido no es muy bueno y que tampoco se ajusta a lo que hemos visto que funciona (contenido de calidad con complejidad baja para que mucha gente pueda entenderlo) podemos ver que el número de suscriptores de los canales

de habla hispana es seguramente más bajo de lo que podría ser en la mayoría de los casos. Realmente no existe ningún canal realmente bueno de habla hispana (aunque el de la Figura 4 es lo que más se le parece). Aquí algunos ejemplos:



Figura 4. Hagamos videojuegos 42466 suscriptores

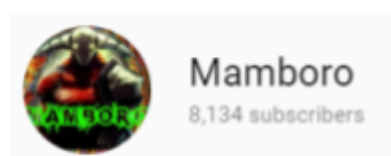


Figura 5. Mamboro 8134 suscriptores

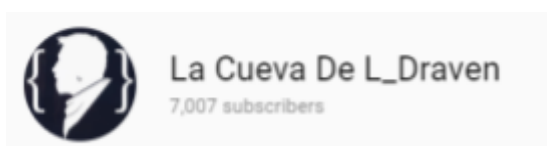


Figura 6. La Cueva De L_Draven 7007 suscriptores

Resumen de los tutoriales de youtube:

| Nombe Canal | Número de Suscriptores | Complejidad del contenido | Calidad del contenido | Idioma del contenido |
|----------------------|------------------------|---------------------------|-----------------------|----------------------|
| Brackeys | 411551 | Baja | Alta | Inglés |
| Sebastian Lague | 118807 | Alta | Alta | Inglés |
| quill18creates | 128802 | Alta | Baja | Inglés |
| Hagamos Videojuegos | 42466 | Baja | Alta | Español |
| Mamboro | 8134 | Media | Normal | Español |
| La Cueva De L_Draven | 7007 | Baja | Normal | Español |

Figura 7: Resumen Youtube

- 2) **Tutoriales escritos:** Por otro lado tendríamos la opción menos popular y menos consumida. Los tutoriales escritos, esos largos artículos y extensas documentaciones que la gente en general no suele consumir. Esta opción es prácticamente inexistente, sobre todo en español. Es la opción que más tiempo requiere por parte del creador y además es la menos popular. Aunque al menos siempre vamos a poder contar con la documentación oficial de Unity [2].

Si buscamos artículos en inglés solemos encontrar por lo menos una entrada que nos pueda servir pero por desgracia casi nunca en español.

- 3) **Cursos de pago:** Aunque la opción más popular, como ya hemos visto, es Youtube. La opción más lucrativa es sin duda producir y poner a la venta un curso de pago con unas 10-20 horas de contenido audiovisual de alta calidad disponible para ver en cualquier momento. Estos cursos se suelen vender en plataformas tales como Udemy, su precio suele ser de unos 15 euros y el número de estudiantes/gente inscrita depende de la inversión en marketing, de lo bien que la página web que hostee el curso lo posicione y obviamente de la calidad del mismo. Aquí algunos ejemplos:



Figura 8. Curso en español más vendido en Udemy



Figura 9. Curso en inglés más vendido en Udemy



Figura 10. Los únicos cursos disponibles en Coursera (muy poco populares)



Figura 11. Curso encontrado en domestika.org

La conclusión que podemos sacar al respecto es que con un buen curso estructurado, de calidad y con un precio razonable hay un amplio mercado dispuesto a comprarlo. También vemos que Udemy parece ser la mejor plataforma donde publicar dichos cursos y vemos también que hay mucha más gente de habla inglesa dispuesta a invertir en este tipo de producto (mirar Figura 9).

- 4) **Foros de desarrolladores:** Son el sitio ideal para resolver dudas, para contactar con otros desarrolladores y para enterarse de eventos que se llevan a cabo en la comunidad. Aquí también contamos con muchas más opciones en inglés que en español. En español solo contamos con una opción (ya que el resto han caído en desuso):
- UnitySpain [3]

En conclusión, después de ver estos 4 métodos de aprendizaje podemos ver que Youtube es la mejor manera de llegar a una gran audiencia y que los cursos son la mejor manera de conseguir ingresos. Las otras dos opciones, los tutoriales escritos y los foros no se usan mucho. Pero hay un pequeño grupo de gente que sí agradece su existencia.

Y es por todos estos motivos que el material docente de la plataforma se hará en forma de vídeo y en forma de texto. El formato vídeo se subirá tanto en Youtube como en la plataforma y se priorizará el contenido de complejidad más simple. Por otro lado el material docente en forma de texto se hará en forma de posts en la plataforma y servirá para acompañar y facilitar el entendimiento del vídeo subido en Youtube que también estará disponible en el mismo post. Como ya hemos visto la plataforma también contará con un foro para resolver dudas y en un futuro, si la plataforma tiene éxito, no se descarta crear un curso de pago y colgarlo en Udemy.

4. Alcance del proyecto

4.1. Meta final

Así pues, cuál es la meta final del proyecto? Con este proyecto se pretende unificar los cuatro aspectos analizados en el estudio de mercado (Youtube, tutoriales escritos, cursos y foro online) bajo una misma plataforma web orientada al desarrollo de videojuegos en Unity, tal y como hemos comentado en el estudio de mercado. Aunque al principio nos centraremos en contenido en forma de vídeo y texto que servirá como soporte del vídeo. Esta plataforma ofrecerá una base de conocimiento sólida y útil para que desarrolladores de habla hispana puedan aventurarse en la programación de videojuegos sin la necesidad de acudir a otros sitios web y se creará usando Wordpress con tal de ahorrar tiempo y poder completar el resto de objetivos a tiempo.

La parte más fundamental del proyecto será la creación de la plataforma que dé soporte a esta idea y si el tiempo lo permite también empezar a producir contenido didáctico en forma de un canal de Youtube, de un curso de pago que habría que promocionar de alguna forma y tutoriales escritos. Hay que decir que todo lo que se hiciera fuera de la plataforma a su vez serviría para redirigir gente hacia la plataforma. Para ello la plataforma web deberá contar con:

- Posibilidad de crear nuevos posts de forma fácil (fácil edición y mantenimiento)
- Creación de usuarios y un sistema de login
- Un foro donde los usuarios puedan interactuar y hacer preguntas
- Algún sistema que permita medir tanto la procedencia del tráfico como su volumen
- Funcionalidades que soporten la toma de decisiones
- Creación y seguimiento de un embudo de conversión.
- Soporte para listas de correo donde se pueda avisar de contenido nuevo, dar algún tipo de información o promocionar algún contenido
- Posibilidad de añadir vídeos de Youtube en los posts (en el caso que se cree un canal para producir contenido didáctico)
- Posibilidad de descargar archivos que se adjunten a los posts

Aparte de crear esta plataforma también habrá que considerar temas de marketing tales como considerar que se envía a través de la lista de correo, como hacer que la plataforma llegue a más gente, como aumentar el índice de retención de los usuarios etc...

4.2. Obstáculos y riesgos

El mayor obstáculo va a ser el tiempo, este proyecto tiene una fecha de entrega fijada y desarrollar una plataforma web y llenarla de contenido es ambicioso. Ya que no

solo hay que asegurar que se crea una web robusta, usable, bien planteada y que tiene herramientas propias de un buen sistema de información sino que también hay que pensar qué enseñar, cómo hacerlo y luego producirlo y editarlo de forma que sea atractivo.

El siguiente obstáculo puede ser la tecnología. Nunca he desarrollado una web, así que cualquier problema puede echarme atrás más tiempo del previsto. Para minimizar las posibilidades que esto llegue a ser un problema se usará WordPress para facilitar y agilizar el desarrollo de la plataforma.

Finalmente podría pasar que la plataforma sea un fracaso absoluto que nadie la use o la visite, ya sea por un mala gestión de las decisiones de marketing, por una falta de contenidos interesantes o porque la plataforma sea poco usable.

5. Metodología

A lo largo de la carrera hemos ido viendo numerosas metodologías para desarrollar software. Desde metodologías tradicionales como *Cascade* otras más avanzadas como el *Rational Unified Process* (RUP).

Pero para poder elegir la metodología que más conviene al proyecto nos tendremos que fijar en la naturaleza del proyecto. En este caso necesitaremos que la plataforma empiece a funcionar lo antes posible con tal de poder hacer pruebas, evaluarla con el director del proyecto y al fin y al cabo mejorarla a medida que crece.

Por este motivo, tal como dice el *agile manifesto* [4], preferiremos tener software funcional lo antes posible que una documentación exhaustiva que tampoco es necesaria. También necesitamos poder responder de forma rápida y efectiva a los cambios que se vayan dando. Queremos poder adaptarnos a cambios de planes que surjan durante el desarrollo y queremos poder iterar rápidamente sobre casos de uso. Por ello será conveniente usar una metodología ágil.

5.1. Scrum modificado

Concretamente usaremos un método inspirado en Scrum pero modificado y adaptado a un equipo de una sola persona. La metodología Scrum requiere de diversos roles para su correcto funcionamiento pero al tratarse de un equipo de una sola persona lo adaptamos de forma que la esencia sea la misma pero no su funcionamiento. Así que lo que haremos será aplicar una metodología ágil inspirada en Scrum.

Para aplicarla se realizarán diversas iteraciones o sprints definidas en el apartado de planificación temporal. En cada una de estas iteraciones se realizarán un número

reducido de historias de usuario y al final de cada iteración se obtendrá una nueva versión del software funcional y por lo tanto una posible versión final apta para demostraciones y testeo.

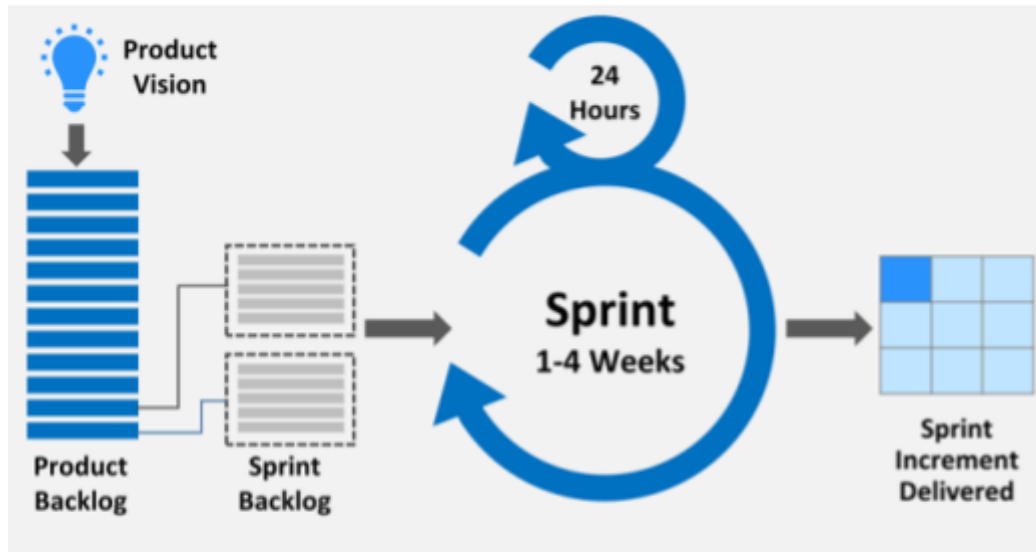


Figura 12. Funcionamiento Scrum

El seguimiento se hará a través de Trello[5] ya que es una herramienta sencilla que a la vez nos permite hacer un muy buen seguimiento de las iteraciones que definamos (estas iteraciones están definidas en el apartado de Tareas dentro de Planificación temporal).

Como podemos ver en la figura 12 las iteraciones o sprints vienen definidos por el backlog del producto o dicho de otra manera las características y funcionalidades que queremos que tenga nuestro producto final. Al principio de cada iteración la dividiremos en diferentes tareas (será el equivalente al sprint backlog de la figura) y las añadiremos a Trello. Luego, para garantizar que las iteraciones se cumplen y se llevan a cabo como deberían cada día se analizará lo que más conviene hacer dentro de las tareas que habremos definido previamente en Trello. Y finalmente, al acabar el día habrá que analizar el trabajo hecho y actualizar Trello en consecuencia, ya sea para añadir tareas nuevas (imprevistos) que necesitemos o marcar como hechas aquellas que hayamos completado.

Esta forma de trabajar nos garantiza total flexibilidad, capacidad de reacción y control del tiempo. Sabemos cuanto tiempo tenemos para cada iteración y las tareas que tenemos que hacer para finalizarla así que en todo momento podremos evaluar como de atrasado o avanzado está el proyecto.

5.2.Métodos de validación

Es esencial testear cada iteración que finalizemos. Hay que asegurar que todos los tests que comprueban cada historia de usuario se pasan y que no hemos roto ninguna funcionalidad de alguna iteración previa.

Para ello, dado que hay pocas funcionalidades en la página web podemos testarlo manualmente. No se requiere de un test suite que haga test unitarios de forma automática ya que perderíamos más tiempo preparando el entorno que el que ganaríamos. No sale a cuenta.

6.Planificación Temporal

La duración de este proyecto está limitada por las fechas de entrega de los TFG. En este caso el proyecto se empezó en febrero y acabará a mitades de junio de 2018.

Para poder acabar el proyecto a tiempo será necesario definir y establecer el orden de las tareas. Posteriormente habrá que decidir cómo vamos a incorporar estas tareas en la metodología establecida y descrita en el anterior punto.

6.1.Tareas

Habrà que realizar las siguientes tareas:

| Tareas | Horas | Rol |
|--|-------|--------------------------------------|
| 0. Fase Inicial | 90 | Jefe de Proyecto |
| 1. Preparar el dominio, servidor y WordPress | 15 | Desarrollador Web |
| 2. Aprender a usar WordPress | 40 | Desarrollador Web |
| 3. Crear Página principal de la plataforma | 10 | Desarrollador Web y Artista |
| 4. Preparar forma de añadir posts a la plataforma con soporte de vídeo y descargas | 15 | Desarrollador Web |
| 5. Search engine optimization (SEO) | 10 | Desarrollador Web |
| 6. Implementar listas Correo y su diseño | 10 | Desarrollador Web y Jefe de Proyecto |

| | | |
|--|-----|--------------------------------------|
| 7. Sistema de cuentas de usuario y login/logout | 20 | Desarrollador Web |
| 8. Parte social de la plataforma: Foro y comentarios en los posts | 25 | Desarrollador Web |
| 9. Herramientas para medir y analizar el tráfico de la plataforma | 15 | Desarrollador Web |
| 10. Creación de posts para la plataforma | 50 | Programador de videojuegos y Artista |
| 11. Creación de vídeos para Youtube que también serán accesibles por la plataforma | 50 | Programador de videojuegos y Artista |
| 12. Creación de un curso de Udemy | 100 | Programador de videojuegos y Artista |
| Total: | 450 | |

Figura 13: Tabla de tareas

Como se puede apreciar en la tabla tenemos una aproximación del tiempo que va a llevar cada tarea. Pero dado la falta de experiencia en el campo del desarrollo web con WordPress los tiempos podrían acabar ser no muy precisos.

Los diferentes colores marcan las diferentes iteraciones o sprints de nuestro Scrum modificado. Estas iteraciones durarán entre 1 semana y 4 y al final de cada una tendremos una versión de la plataforma lista para desplegar. En el siguiente punto podemos ver un mejor desglose de todas estas tareas.

Descripción de las iteraciones:

1. Se trata de conseguir que la plataforma esté online y tenga una landing page. Para ello habrá que comprar el dominio en google domains, el plan personal de Wordpress (ya que es el más conveniente para este proyecto) y habrá que pagar el hosting que se decida.

Cuando esté todo configurado aprenderé a usar Wordpress. Para ello tengo localizado un curso con muy buenas reseñas en Udemy. Una vez adquirida un conocimiento base de Wordpress ya estaré listo para crear la página principal.

2. En la segunda iteración crearemos todos los aspectos básicos de la plataforma. Esto incluye un sistema que nos permita añadir posts/tutoriales de forma fácil, hacer la optimización para motores de búsqueda y crear una lista de correo a través de la cual podamos informar a los usuarios sobre las novedades de la página.
3. Una vez tengamos la base vamos a añadir todos los aspectos sociales y más complejos de la plataforma como ahora el foro, el sistema de usuarios y formas de medir y analizar el tráfico de la web.

4,5,6. El resto de iteraciones consisten en crear contenido docente de calidad ya sea para colgarla directamente en la plataforma o fuera de ella. Aunque en todos los casos este contenido va a servir para atraer a gente a la plataforma.

6.2.Diagrama de Gantt

El diagrama de Gantt resultante de la estimación de horas para cada tarea realizado en el apartado anterior es el siguiente:

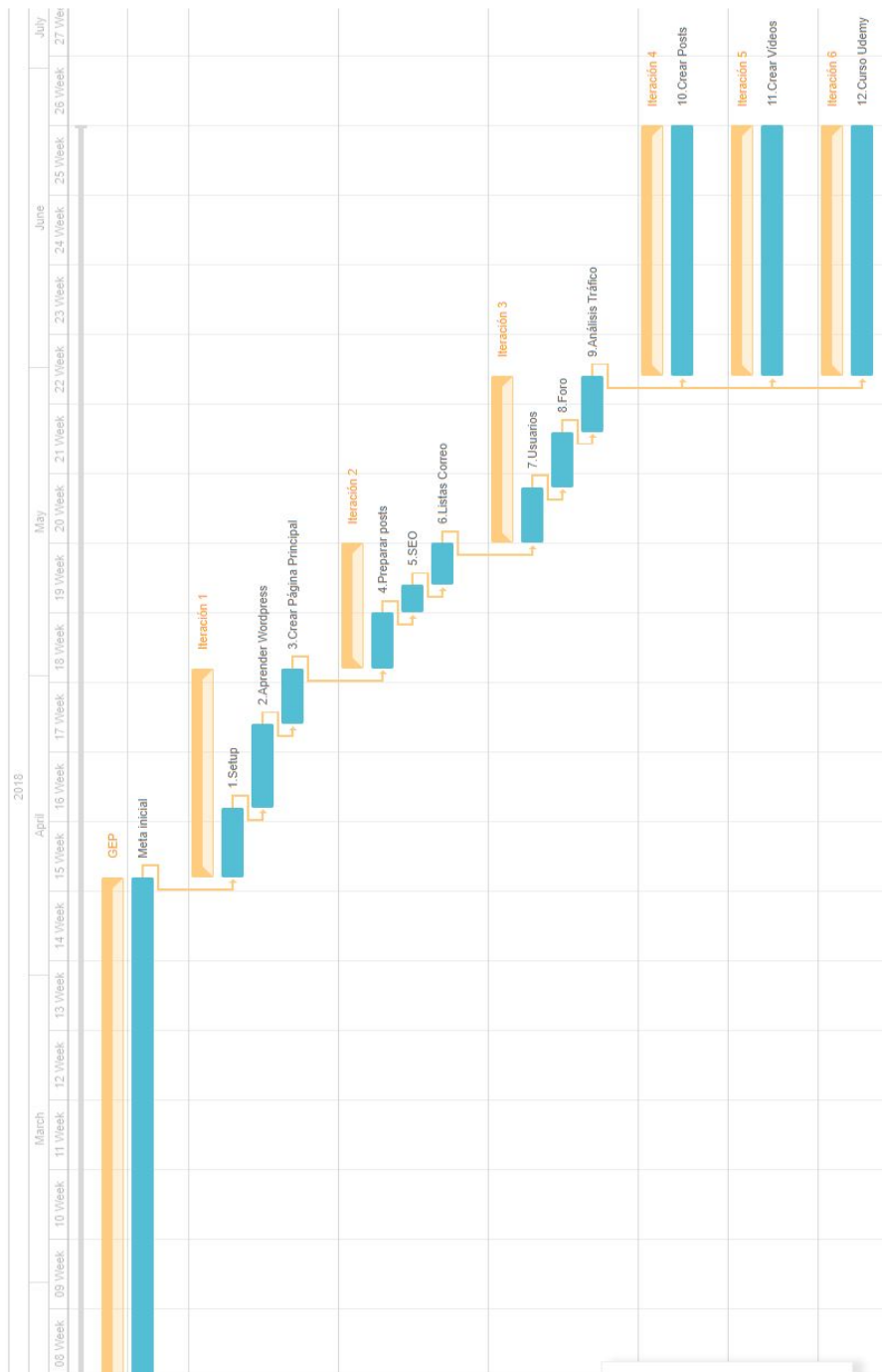


Figura 14: Diagrama de Gantt

Quisiera destacar que solo va a trabajar una sola persona en este proyecto. En este caso yo, Gerard. Y la jornada de trabajo será de unas 4-5h. Este será el tiempo diario invertido en el proyecto. En este tiempo tendré que asumir uno o varios roles de los vistos en el apartado de recursos humanos que podemos encontrar a continuación.

También me gustaría destacar que el motivo de que las últimas 3 iteraciones estén solapadas es que a fin de cuentas engloban la creación de contenido y esta seguramente se acabe haciendo en paralelo. Estas tareas engloban más horas de las que una sola persona puede abarcar en el tiempo que dura el proyecto así que seguramente habrá que priorizar una/s de estas tareas más que otras con tal de tener el máximo de contenido posible antes de la fecha de entrega.

6.2.1.Dependencias

Si nos fijamos en las dependencias del diagrama de Gantt podemos ver que cada tarea depende de la anterior. Esto es así porque solo contamos con un solo trabajador para el proyecto y además en la mayoría de los casos no podemos realizar una tarea a menos que hayamos realizado la que le precede. Por ejemplo no podemos añadir funcionalidades a la plataforma si ni siquiera hemos preparado el dominio y el hosting.

7.Recursos

Para realizar el proyecto propuesto se necesitarán diferentes recursos, tanto de hardware como de software.

7.1.Recursos Humanos

Para realizar el proyecto se va a necesitar una persona o varias que puedan cubrir los siguientes roles:

- Jefe de Proyecto
- Desarrollador web (preferiblemente con experiencia en Wordpress)
- Artista para crear tanto el arte de la web como para los juegos que crearemos para los tutoriales
- Programador con experiencia en Unity y C#

7.2.Hardware

Los recursos de hardware que se van a usar son:

- Ordenador potente para poder desarrollar videojuegos de todo tipo y producir contenido audiovisual de forma rápida, eficiente y sin problemas de rendimiento. Estos son sus componentes:
 - AMD Ryzen 5 1600X 3.6GHz

- NVIDIA GTX 1070 8GB
- 16GB DDR4 RAM
- 120GB SSD
- 2TB 7200RPM HDD
- Placa base y fuente de alimentación acorde al resto de componentes.

7.3. Software

Los recursos de software que se van a usar son:

- Trello: Seguimiento del Scrum modificado
- WordPress: Es un open-source content management system usado para crear páginas web. Además es gratuito y open source siempre que nuestra página web Wordpress esté en un servidor propio. Como es nuestro caso.
- Unity: El motor de videojuegos (game engine) sobre el que se basará todo el material docente que generemos.
- Visual Studio: El entorno de desarrollo que usaremos para programar para Unity.
- Adobe Photoshop y Illustrator: Para generar todo el arte que necesitemos para la plataforma y el material docente.
- GitHub: Usaremos Git para el control de versiones.

8. Valoración de alternativas y plan de acción

En el apartado de obstáculos y riesgos vimos que se podrían dar 3 problemas: falta de tiempo, problemas con la tecnología, fracaso de la plataforma.

8.1. Falta de tiempo

En caso de una falta de tiempo habría que tomar medidas drásticas. El primer paso sería añadir las tareas pendientes a la iteración actual. Pero en caso de que esto no fuera viable o se acumularan varias tareas a lo largo de las iteraciones habría que tomar la decisión de que recortar y de que podemos prescindir.

Esto no supondría un problema grave dado a la metodología ágil que usamos y a la naturaleza del proyecto. El proyecto es muy modular, tiene muchas partes independientes que no interactúan entre ellas y además el contenido docente que podemos añadir a la plataforma es infinito pero nuestro tiempo no lo es. Siempre cabe la opción de recortar un poco en material docente y acabar correctamente la plataforma.

La otra opción sería alargar el proyecto unas semanas después de entregar el TFG. En este caso esto supondría un aumento de los costes ya que habría pagar el sueldo del trabajador, en este caso yo y unas dos semanas más de tiempo.

8.2.Problemas con la tecnología

Este sería sin duda el problema más grave. Si una de las funcionalidades pensadas para la plataforma costara mucho de hacer, diera muchos problemas o necesitara mucho más tiempo del disponible seguramente se tendría que acabar descartando y dejando para después del TFG, lo cual aumentaría el coste del proyecto.

Lo más difícil de implementar seguramente va a ser la unificación de todos los casos de uso bajo el mismo sistema de login de usuarios. Ya que a pesar de que existen soluciones en forma de plugins de Wordpress para todo lo que tenemos en mente habrá que ver si podemos atar todas estas funcionalidades entre ellas aprovechando el sistema de usuarios que implementaremos.

8.3.Fracaso de la plataforma

Sería decepcionante pero no un problema grave. Si por algún motivo la plataforma se acaba no usando y no hubiera tráfico de usuarios implicaría que bien los contenidos o el enfoque del proyecto no han sido buenos.

Pero la calidad del proyecto será la misma con muchos usuarios o con pocos, tanto para bien como para mal. La calidad será objetiva e independiente al tráfico de la plataforma.

8.4.Otros

En caso de que surgiera otro problema no previsto podríamos pivotar y adaptarnos a los cambios gracias a la metodología ágil que usamos. Esto lo haríamos modificando los tiempos y objetivos de las siguientes iteraciones a la actual y no cambiaría los costes del proyecto ni el tiempo necesario para acabarlo pero igual sí que haría que el producto final diferiera de nuestra visión inicial.

9.Presupuesto y sostenibilidad

Este proyecto, igual que cualquier otro, va a desencadenar una serie de consecuencias que causarán un impacto en el mundo en el que vivimos. Estas consecuencias serán diversas y de mucha o poca importancia pero muy importantes a tener en cuenta. En este apartado vamos a identificar y analizar estas consecuencias con tal de ver cómo de sostenible es este proyecto.

9.1.Autoevaluación competencia de sostenibilidad

El primer paso será hacer una autoevaluación sobre mis nociones de sostenibilidad sobretudo en relación a los proyectos TIC, pero no exclusivamente. Esto es importante para tener una referencia de cual es el punto de partida y el punto de vista desde el cual se va a analizar la sostenibilidad del proyecto.

Para hacer esta tarea los profesores de GEP nos han mandado completar una encuesta que da que reflexionar. A estas alturas todos somos más o menos conscientes del impacto ambiental, económico y social que tienen las nuevas tecnologías aunque muchas veces se nos olvida o lo damos por sentado. Por eso lo ideal sería que todo proyecto TIC considerara el impacto que va a tener y se asegurara de cumplir los siguientes requisitos:

- Minimizar un impacto medioambiental negativo.
- Ser consciente del rol que juegan las TIC en la sostenibilidad del planeta.
- En la medida de lo posible maximizar la reutilización de recursos.
- Valorar y maximizar el grado de accesibilidad, la calidad ergonómica, el nivel de seguridad.
- Contribuir al el bien común.
- Introducir la justicia social, equidad, diversidad, y la transparencia al proyecto.

Obviamente esto sería lo ideal, pero lo ideal pocas veces es posible. Sin ir más lejos yo mismo me considero aún poco educado y concienciado con respecto a la sostenibilidad y tengo aún muchos puntos débiles al respecto, aunque también tengo puntos fuertes:

- Puntos débiles:
 - Aunque en mi especialidad (sistemas de información) hemos tocado varios temas de economía de empresa y del impacto económico de un proyecto creo que aún tengo un conocimiento algo precario del tema y no sería capaz de hacer un análisis económico completo de todo un proyecto complejo.
 - En mis anteriores proyectos nunca he considerado el impacto medioambiental, económico o social de lo que hacía a pesar de ser capaz de hacerlo.

- No conozco los principios deontológicos relacionados con la sostenibilidad en un proyecto TIC.
- Puntos fuertes:
 - Soy capaz de explotar y estoy familiarizado con todo tipo de herramientas de trabajo colaborativo del ámbito de las TIC.
 - Conozco los conceptos de creatividad e innovación y estrategias para desarrollarlos y puedo aportar ideas para hacer un proyecto TIC más sostenible.
 - Comprendo los costes ambientales que tienen los productos relacionados con las TIC a lo largo de su ciclo de vida. Estoy familiarizado con el desarrollo de productos software.

9.2. Matriz de sostenibilidad

La matriz de sostenibilidad presenta el resultado de la evaluación de este proyecto durante la fase inicial, o sea: durante el periodo de GEP y por lo tanto antes de empezar el desarrollo del proyecto. Esta matriz está dividida en 3 grandes bloques: **Proyecto Puesto en Producción (PPP)**, **Vida Útil** y **Riesgos**. Para cada uno de estos bloques se ha realizado una valoración del impacto ambiental, económico y social y además se le ha asignado un valor entre 0 y 100 que indica cómo de sostenible es cada aspecto del proyecto, siendo 100 un indicador de sostenibilidad máxima.

| | PPP | Vida Útil | Riesgos |
|-------------------------|--------------------|--------------------|---------------------|
| Ambiental | Consumo del diseño | Huella ecológica | Riesgos ambientales |
| | 95/100 | 90/100 | 90/100 |
| Económico | Factura | Plan de viabilidad | Riesgos económicos |
| | 60/100 | 50/100 | 60/100 |
| Social | Impacto personal | Impacto social | Riesgos sociales |
| | 75/100 | 75/100 | 95/100 |
| Rango de sostenibilidad | 230/300 | 215/300 | 245/300 |
| | 230/300 | | |

Figura 15: Matriz de sostenibilidad

Podemos ver que el rango de sostenibilidad total es 245 (la casilla roja de la Figura 15) de un máximo de 300 lo que significa que considero que el proyecto es altamente sostenible. Para más detalles y el porqué de estas valoraciones consultar los siguientes subapartados.

9.2.1. Impacto ambiental

Tal como vimos en el apartado de Actores Implicados este proyecto cuenta con un solo desarrollador. Así que para estimar el impacto ambiental del proyecto durante su desarrollo solo tendremos que considerar el impacto ambiental que tienen las herramientas que va a usar dicho desarrollador a lo largo del proyecto.

En este caso el desarrollador soy yo mismo, Gerard. Y para desarrollar el proyecto voy a usar mi ordenador de sobremesa de gama media/alta ya que me va a garantizar poder desarrollar todos los aspectos del proyecto de forma fluida y sin ningún tipo de problema de rendimiento.

Para medir el consumo de este ordenador he usado una herramienta online que calcula el consumo de un ordenador dadas sus especificaciones y una descripción del uso que se le va a dar [6]. En este caso dado la herramienta me ha dicho que el ordenador consumiría 83W de media cuando esté activo y 3W cuando esté apagado. Entonces si consideramos la Planificación Temporal ya vista nos sale:

Consumo total PC encendido: $83W * 450h = 37.35 \text{ kWh}$
Consumo total PC apagado: $3W * 1400h = 3.99 \text{ kWh}$
Consumo total: 41.34 kWh

Considerando que el consumo medio de una casa en un mes es de unos 300 kWh podemos evaluar que el desarrollo de este proyecto va a consumir muy poca energía en relación a lo que consumimos ya de por sí durante nuestro día a día.

Aparte de el PC que se usará para el desarrollo también deberíamos considerar el consumo del servidor que va a hostear la plataforma web ya que consumirá mucha más energía que el PC. Dado que el volumen de accesos, procesamiento e información que tendrá que gestionar la plataforma será muy variable en el tiempo optamos por una infraestructura de servidores elástica. Esto significa que los recursos que se asignan al hosteo de este proyecto cambiarán según la demanda que haya en ese preciso instante. Al elegir esta opción garantizamos que el consumo, a pesar de no saber cuál será, es lo más bajo posible y que solo gastamos los recursos que nuestro proyecto realmente necesita.

Por todos estos motivos hemos puntuado muy positivamente el impacto ambiental en la matriz de sostenibilidad [Figura 15]. Por un lado el consumo del proyecto puesto en producción (PPP) será muy bajo y el servidor solo consumirá lo que realmente

necesite. Por otro lado el impacto por la vida útil y riesgos del proyecto será también bajo ya que lo peor que puede pasar es que el tiempo de desarrollo se alargue más de la cuenta o que el volumen de gente que use la plataforma sea tan alto que se dispare el consumo energético. Hay que remarcar que a pesar de que el coste energético de este proyecto es bajo esta solución no es mejor que otras ya existentes y usadas por los competidores.

9.2.2.Impacto Económico

Para medir el impacto económico de este proyecto lo primero que haremos será medir los costes del mismo. Para ello los vamos a dividir en las categorías de un típico proyecto TIC (tanto el hardware como el software están listados en el apartado de recursos):

1. **Recursos humanos:** Aunque este proyecto será llevado por una sola persona existen varios roles que esta tendrá que asumir. A continuación se detallan las horas realizadas por cada rol y el dinero destinado a cada rol:

| Rol | Sueldo por hora | Tiempo estimado | Coste estimado |
|----------------------------|-----------------|-----------------|----------------|
| Jefe de Proyecto | 30€/h | 100h | 3000€ |
| Desarrollador Web | 20€/h | 145h | 2900€ |
| Artista | 25€/h | 55h | 1375€ |
| Programador de videojuegos | 15€/h | 150h | 2250€ |
| TOTAL | | | 9525€ |

Figura 16: Sueldos por rol

2. **Hardware:** En cuanto a hardware vamos a necesitar un ordenador potente ya descrito en el apartado de recursos.

| Hardware Necesario | Precio | Amortización (considerando que sólo se usará durante el proyecto) |
|--------------------|--------|---|
| Ordenador | 1300€ | $1300€/450h = 2.8€/h$ |

Figura 17: Presupuesto Hardware

3. **Software:** En cuanto a software vamos a tener que comprar Adobe Photoshop y Adobe Illustrator. El resto de software (ya descrito en el apartado de recursos) es gratuito.

| Software Necesario | Precio | Amortización (considerando que sólo se usará durante el proyecto y por el rol de artista) |
|--------------------|-------------|---|
| Adobe Photoshop | 184€ | $184\text{€}/55\text{h} = 3.34\text{€/h}$ |
| Adobe Illustrator | 187€ | $187\text{€}/55\text{h} = 3.4\text{€/h}$ |
| TOTAL | 371€ | 6.74€/h |

Figura 18: Presupuesto Software

4. **Despesas generales:** Habrá que pagar la electricidad usada para el desarrollo, así que si el precio medio de la electricidad en España es de 0.145€ por kWh nos costará 6€. También tendremos que pagar el hosting de la página web que costará aproximadamente 7.5€ al mes y el plan personal de Wordpress que permite usar un dominio propio y elimina los anuncios para los visitantes de la web. Este plan personal vale 4€ al mes. Por último habrá que pagar al proveedor de internet, en este caso Jazztel que nos ofrece fibra óptica de 400mb simétrica a 42.95€ al mes.
5. **Otras despesas:** Para aprender a usar Wordpress usaré un curso de Udemy de 12€ con muy buenas reseñas.
6. **Imprevistos:** En esta sección se van a considerar el coste de los obstáculos y riesgos. Estos son:
 1. Avería de algún componente del ordenador: Requeriría de acción rápida para no retrasar el proyecto. Seguramente causaría un parón de un día ya que me obligaría a identificar el problema, salir a comprar un sustituto de la pieza en mal estado y luego reemplazarla. En el peor de los casos el coste de la avería sería de 550€ si es la gráfica el componente que se estropea. Aunque la probabilidad de que esto ocurra es muy baja, le asignaremos una probabilidad del 5%.
 2. Retraso de la implementación o creación de contenido: En el caso que nos retrasamos 15 días, cosa bastante probable y que por ello le asignaremos una probabilidad del 25%, habría que considerar las horas extras de trabajo. Así que si nos retrasamos 15 días, el sueldo medio de los roles de nuestro proyecto es de 22.5€ y trabajo 5 horas al día el coste de este retraso sería: $15 \text{ días} \times (22.5\text{€} \times 5\text{h}) = 1687.5\text{€}$.

| Imprevisto | Probabilidad | Precio | Coste |
|--------------------|--------------|---------|----------|
| Avería Ordenador | 5% | 550€ | 27.5€ |
| Retraso de 15 días | 25% | 1687.5€ | 421.875€ |

| | |
|-------|----------|
| TOTAL | 449.375€ |
|-------|----------|

Figura 19: Costes de imprevistos

7. **Contingencias:** Se reserva también algo de dinero por si surge algún problema durante el desarrollo del proyecto que hace aumentar el coste del mismo. En este caso será un 10% de nuestro presupuesto.

Control de costes: Habrá que hacer un control del dinero que se va invirtiendo en el proyecto con tal de comparar y evaluar las desviaciones entre el presupuesto y los costes reales en el caso que las haya.

Para hacerlo analizaremos los costes y las inversiones al final de cada iteración y en caso de que se dé una desviación será esencial identificar donde se ha producido la desviación, el porqué y la cuantía del desvío. Una vez identificado estos factores podremos, en el caso que sea necesario modificar la siguiente iteración del proyecto conforme a lo que creamos más conveniente.

Conclusiones:

| Concepto | Precio |
|--------------------|--------|
| Recursos humanos | 9525€ |
| Hardware | 1300€ |
| Software | 371€ |
| Electricidad | 6€ |
| Hosting | 90€ |
| Wordpress personal | 48€ |
| Internet | 515€ |
| Curso Udemy | 12€ |
| Imprevistos | 449€ |
| Contingencias | 1231€ |
| TOTAL | 13547€ |

Figura 20: Desglose coste total

Con este coste total podemos afirmar que el precio del proyecto no es alto si fuera una empresa la que hace la inversión inicial, pero sin embargo sí sería una inversión considerable para un particular. Así pues podemos justificar los valores asignados al impacto económico en la matriz de sostenibilidad [Figura 15] que precisamente es donde más flaquea el proyecto. Vemos que tanto el dinero necesario para el proyecto puesto en producción (PPP) y el riesgo es bastante alto ya que dado el modelo de negocio de la plataforma (todo gratuito en un principio) hace que la inversión inicial sea considerable. Y la vida útil de este proyecto aún está por ver, no sabemos si la plataforma será exitosa o no y de hecho es uno de nuestros factores de riesgo. Por eso lo hemos puntuado con un 50/100.

9.2.3. Impacto Social

Para medir el posible impacto social de este proyecto empezaré hablando del impacto que este creo que va a tener en mí. Llevo tiempo queriendo crear una página web pero nunca encontraba el momento, pero por suerte ahora tengo la oportunidad perfecta para hacerlo. Por otro lado siempre me ha gustado compartir mis conocimientos y disfruté muchísimo de un curso de desarrollo de videojuegos en Unity que impartí el invierno de este año. Así que crear este proyecto me será muy útil para aprender a desarrollar webs, ganar soltura en cuanto a mis habilidades docentes y también ver si podría llegar a dedicarme a este campo o no.

Además de afectarme a mí, este proyecto, con suerte, afectará sobretodo a todos los usuarios que usen la plataforma web resultante. Estos usuarios que en su mayoría serán jóvenes con poca experiencia en programación y con ganas de crear videojuegos acudirán a la web con la esperanza de aprender a hacer juegos con Unity. Así que si mi método didáctico es acertado y consigo transmitir correctamente mi pasión y entusiasmo por la materia estos alumnos podrían verse altamente beneficiados por la plataforma. E incluso en un futuro, si descubren que desarrollar videojuegos es su vocación podría llegar a cambiar el transcurso de sus vidas profesionales.

Esta plataforma como hemos comentado pretende unificar en un solo sitio conocimiento sobre desarrollo de videojuegos y presentarlo de la mejor forma posible facilitando la asimilación de los conceptos presentados. Por eso opino que existe una necesidad real de este proyecto y que si se realiza bien puede afectar positivamente a muchas personas.

10. Seguimiento

Todo lo que hemos visto hasta este punto es la planificación y las predicciones. Todo lo anterior a este apartado se escribió y pensó antes de empezar con el desarrollo. Y como todo informático sabe las cosas nunca acaban de salir del todo como uno espera. Así pues este apartado pretende ilustrar el seguimiento que se hizo a lo largo del proyecto y ver hasta qué punto nuestras predicciones iniciales eran acertadas, que desviaciones o cambios han habido y el impacto que han causado en el proyecto.

Como hemos visto previamente en el apartado de metodología elegimos utilizar un método scrum modificado para una sola persona (ver el apartado 5.1 para más información) con el objetivo de ser ágiles, poder trabajar rápido y ser capaces de responder a cualquier imprevisto que surgiera.

Y realmente ha funcionado a las mil maravillas. El hecho de tener que definir todas las tareas al principio de cada iteración, ponerlas en Trello y luego hacer un seguimiento diario casi religioso me ha permitido seguir la planificación inicial, la cual por cierto era muy acertada.

10.1. Desviaciones

Aunque como era de esperar han habido pequeñas desviaciones. En primer lugar las horas que había calculado para las distintas tareas de los distintos apartados no ha sido del todo exacta. Algunas tareas han llevado más tiempo del previsto mientras que otros apartados han requerido menos. Pero al final se ha compensado bastante y la distribución de horas final es muy parecida a la que hice inicialmente.

Por otro lado, el cambio de planificación más grande vino después de hablar con mi tutor Marc Alier. Tras nuestra conversación me he dí cuenta que tenía que cambiar un poco el enfoque del proyecto, en vez de intentar crear mucho contenido didáctico para la plataforma es mucho más interesante desde el punto de vista de un proyecto de Sistemas de Información fortalecer y ampliar las herramientas y mecanismos que me ayudarán a tomar decisiones, a popularizar la web, optimizar el contenido, obtener leads y crear un buen embudo de conversión. Así pues modifiqué las últimas 3 tareas del proyecto que antes eran:

| | | |
|--|----|-----------------------|
| 10. Creación de posts para la plataforma | 50 | Programador y Artista |
|--|----|-----------------------|

| | | |
|--|-----|--|
| 11. Creación de vídeos para Youtube que también serán accesibles por la plataforma | 50 | Programador y Artista |
| 12. Creación de un curso de Udemy | 100 | Programador y Artista y jefe de proyecto |

Figura 21: Últimas 3 tareas antes de los cambios

Mientras que las tareas que se han acabado realizando y la distribución de horas final ha sido:

| Tareas | Horas | Rol |
|--|-------|--------------------------------------|
| 0. Fase Inicial | 90 | Jefe de Proyecto |
| 1. Preparar el dominio, servidor y WordPress | 15 | Desarrollador Web |
| 2. Aprender a usar WordPress | 40 | Desarrollador Web |
| 3. Crear Página principal de la plataforma | 10 | Desarrollador Web y Artista |
| 4. Preparar forma de añadir posts a la plataforma con soporte de vídeo y descargas | 10 | Desarrollador Web |
| 5. Search engine optimization (SEO) | 10 | Desarrollador Web |
| 6. Implementar listas Correo y su diseño | 20 | Desarrollador Web y Jefe de Proyecto |
| 7. Sistema de cuentas de usuario y login/logout | 25 | Desarrollador Web |
| 8. Parte social de la plataforma: Foro y comentarios en los posts | 25 | Desarrollador Web |
| 9. Herramientas para medir y analizar el tráfico de la plataforma | 15 | Desarrollador Web |

| | | |
|---|-----|--------------------------------------|
| 10. Creación de posts para la plataforma y algún vídeo de Youtube que también será accesibles por la plataforma | 50 | Programador de videojuegos y Artista |
| 11. Integración de Google Analytics y sistema de toma de decisiones | 20 | Desarrollador Web y Jefe de Proyecto |
| 12. Presencia en redes sociales | 25 | Jefe de Proyecto |
| 13. Optimizar la landing page al máximo | 15 | Jefe de Proyecto |
| 14. Diseñar un buen embudo de conversión | 30 | Jefe de Proyecto |
| Total: | 400 | |

Figura 22: Tabla de tareas final

10.2.Impacto

Estos cambios, como es de esperar, han tenido cierto impacto en algunas cosas previamente dichas. Como por ejemplo la iteración 4 (la verde) ha sido mucho más corta y se ha añadido una nueva iteración. La nueva iteración 5 que cubre muchos de los aspectos que han potenciado al proyecto como el Sistema de Información que es. También ahora el proyecto va a necesitar 10h menos y por lo tanto cambiarán los costes del proyecto. Aquí la nueva tabla de costes de recursos humanos:

| Rol | Sueldo por hora | Tiempo estimado | Coste estimado |
|----------------------------|-----------------|-----------------|----------------|
| Jefe de Proyecto | 30€/h | 180h | 5400€ |
| Desarrollador Web | 20€/h | 165h | 3300€ |
| Artista | 25€/h | 30h | 750€ |
| Programador de videojuegos | 15€/h | 25h | 375€ |
| TOTAL | | | 9825€ |

Figura 23: Tabla de recursos humanos final

Así pues los costes de recursos humanos han pasado de 9525€ a 9835€. Y por lo tanto el nuevo coste total del proyecto pasa a ser 138547€ contra los 13547€ iniciales.

11.Implementación de la página

Antes de poder hablar sobre la gestión del sistema de información en el que se ha convertido Aprender Unity tenemos que ver cómo hemos creado la plataforma web y cómo usando nuestra visión inicial hemos podido hacer tangible nuestra misión final.

Desarrollar e implementar todas las funcionalidades que habíamos planeado ha requerido de tiempo, una buena planificación y agilidad a la hora de resolver problemas que han ido surgiendo. Así pues, en esta sección, nos centraremos en todo lo relacionado con el proceso de creación de la plataforma.

11.1.Preparación del dominio y servidor

A la hora de crear una página web en Wordpress tenemos dos opciones:

1. Usar un subdominio de Wordpress (.wordpress.com) hosteado por ellos mismos. Pero que a la vez tiene funciones limitadas.
2. Comprar un dominio, alquilar o crear un servidor y finalmente instalar Wordpress en el servidor.

La gran y única ventaja de la primera opción es lo fácil que es hacer la configuración inicial. Simplemente hay que entrar en wordpress.com, crear una cuenta, elegir el nombre para nuestra página y ya tendríamos una web accesible para todo el mundo, fácilmente editable y además sin pagar nada, así de fácil! Entonces cuál es el problema?

El problema de esta opción son los siguientes: a menos que paguemos wordpress pondrá anuncios en nuestra página, tendremos un subdominio de Wordpress.com, cantaremos con muy poco espacio de disco. Y aunque paguemos, y este es el mayor inconveniente, no podemos instalar ningún plugin (pieza de software que contiene un grupo de funciones que se pueden añadir a una página web Wordpress), muchos temas² no están disponibles y además no podemos modificar absolutamente nada del código de Wordpress ni de los pocos temas que se nos permita descargar.

Así pues esta primera opción es totalmente inviable para este caso ya que nos ataría de manos a pies y seríamos incapaces de crear todas las funcionalidades que

² Un tema es un conjunto de ficheros que trabajan juntos para producir la interfaz gráfica de tu página web. Básicamente es una gran plantilla que define el aspecto de los elementos de tu web. Los temas de Wordpress también suelen incluir funcionalidades y herramientas únicas.

habíamos planteado. Entonces vamos a rápidamente indicar cuales son los pasos que hemos tomado para configurar un servidor propio (la segunda opción).

El primer paso será elegir que servidor usar para hostear la página. Como el tiempo de este proyecto es muy limitado la opción que nos permita configurar todo de la forma más rápida posible (ajustándose así a nuestra planificación inicial) y que a la vez tenga un precio razonable será la que más nos interese. Así pues quedan descartadas todas aquellas que requieran instalación de Wordpress manual como sería montar un servidor en casa, Amazon Web services, Google Cloud etc... Siendo así los proveedores de hosting a elegir Bluehost, HostGator y SiteGround ya que todos ofrecen instalación de Wordpress con un click, gestión a tu espacio de disco en el servidor, posibilidad de configurar correos electrónicos con tu dominio (ahora a continuación hablaremos del dominio) y muchas más opciones. El factor diferencial y por el cual acabé adquiriendo 12 meses de hosting en Siteground fue por un lado el precio que era ligeramente menor y un estupendo servicio de atención al cliente donde te resuelven dudas al momento y que he acabado usando en dos ocasiones. El precio de cada mes es 3.95€+IVA así que el precio total por un año de hosting es 47.4+IVA. Esta es la tarifa más económica pero realmente no necesitamos más, sobretodo ahora que estamos intentando arrancar la web. En un futuro, si la web creciera igual necesitaríamos mejorar nuestro plan mensual, pero en SiteGround siempre estamos a tiempo de pagar la diferencia entre lo que hemos pagado y el plan que deseamos y así mejorar las prestaciones del servidor. Así que empezar con el plan más barato es una apuesta segura. En todo caso estos son los diferentes planes que se nos ofrecen y sus características más esenciales:

| | | |
|---|--|---|
|  StartUp El mejor comienzo con WordPress |  GrowBig Potencia extra para tu WordPress |  GoGeek Diseñado para auténticos geeks |
| PRECIO ESPECIAL 3.95 €/mes. Precio normal 9.95 €/mes. IVA NO INCL. | PRECIO ESPECIAL 6.45 €/mes. Precio normal 17.95 €/mes. IVA NO INCL. | PRECIO ESPECIAL 11.95 €/mes. Precio normal 29.95 €/mes. IVA NO INCL. |
| <ul style="list-style-type: none">✓ Un sitio web✓ 10GB de espacio web✓ Indicado para ~ 10.000 visitas/mes | <ul style="list-style-type: none">✓ Varios sitios✓ 20GB de espacio web✓ Indicado para ~ 25.000 visitas/mes | <ul style="list-style-type: none">✓ Varios sitios✓ 30GB de espacio web✓ Indicado para ~ 100.000 visitas/mes |

Figura 24: Planes Wordpress en SiteGround

Una vez tenemos alquilado nuestro servidor con las características que nos permitirán hostear nuestra página Wordpress tendremos que comprar el dominio para que la

gente pueda llegar a nuestra página web fácilmente. Este proceso es muy sencillo. Simplemente fui a domain.com elegí un nombre de dominio que me gustara y que valiera 10\$ y lo compré durante un año. El dominio elegido fue aprenderunity.com. Una vez tenemos el dominio comprado tenemos que cambiar los dns del dominio para que apunten a nuestro servidor. Al hacerlo tendremos que esperar unos 3 días para que nuestra web sea accesible desde cualquier navegador de cualquier ordenador del mundo debido a que tenemos que esperar a que los cambios de DNS se propaguen al resto de servidores DNS.

11.2.Aprender Wordpress

Llegados a este punto ya tendríamos una página web bien configurada y accesible. Pero hay un problema, el contenido de la web es el contenido de muestra que crea Wordpress al ser instalado. La solución: Aprender como funciona Wordpress, crear un plan de acción y ceñirse a la planificación!

Cuando empecé este proyecto no tenía ninguna noción previa de Wordpress ni de desarrollo web así que para adquirir los conocimientos que más adelante necesitaría para desarrollar Aprender Unity tuve que partir desde cero. Por eso busqué tutoriales y guías para novatos que a la vez se adecuaban a lo que quería conseguir. Tras una búsqueda rigurosa y tras descartar muchas opciones me incliné por tres tutoriales de Youtube de unas dos horas cada uno donde se sigue el proceso de creación de una web completa desde cero [7] [8] [9]. Para aplicar los conocimientos de estos vídeos fui siguiendo y aplicando los ejemplos y procesos que se mostraban. Al acabar tenía una noción amplia de todas las funcionalidades básicas de Wordpress dado que pude ver y replicar todos los conceptos claves como la creación de páginas, como crear una portada atractiva, como generar posts, instalar plugins, crear una barra de menú en todas las páginas etc.

La ventaja de seguir 3 tutoriales en vez de solo uno es que aprendes a usar más herramientas, ves más formas de hacer las cosas y aprendes a usar diferentes temas que aparte de cambiar el aspecto de la web muchas veces también ofrecen funcionalidades y opciones diferentes. En resumen obtienes más recursos y una visión más amplia para trabajar mejor en tu propio proyecto. Al fin y al cabo cuantas más opciones tengamos mejor.

Y el problema que tuvo haber seguido 3 tutoriales distintos es que una vez intenté fusionar herramientas y formas de hacer de distintos tutoriales y no acabó de funcionar. Pero esto es anecdótico. Hacer los 3 tutoriales realmente fue una muy buena decisión.

11.3.Diseño de la página

Teniendo ya todos los conocimientos básicos necesarios era momento de empezar a diseñar la página. Para recapitular un poco Aprender Unity cuenta con los siguientes apartados claramente divididos:

- Landing page / Portada
- Tutoriales / Posts escritos
- Foros

Por tanto nuestro diseño tenía que integrar estos 3 apartados de la mejor forma posible, hacerlos accesibles, fáciles de usar, atractivos y que dieran una buena experiencia de uso al usuario. También me gustaría remarcar que cuando hablamos del diseño de una página web estamos hablando sobretodo de la elección, disposición y posicionamiento de los elementos que componen la página. Después elegir un texto o imagen acertada, una buena paleta de colores o una fuente atractiva aunque es extremadamente importante es secundario. Lo realmente esencial, el núcleo del diseño es la elección, posicionamiento y elección del tamaño de los elementos.

Para garantizar todas estas condiciones me inspiré en muchas páginas web, en normas de diseño básicas que se pueden encontrar en internet [11] y que hemos visto en asignaturas como IDI y por supuesto usando el sentido común e intuición.

Durante todo este proceso de diseño fui tomando notas e haciendo bocetos de cada una de las páginas. El apartado que más trabajo dio es obviamente la portada. Esta será (normalmente) la primera página que vean los usuarios y es crucial que sea atractiva y meticulosamente estructurada. El orden en que se muestran las cosas y la presentación son clave. Hemos de maximizar la retención de visitantes a nuestra web y el diseño de la portada o landing page es nuestra primera herramienta para hacerlo. El diseño y estructura final se puede ver en aprenderunity.com. La idea es dar un fuerte impacto visual inicial y poco a poco ir guiando al usuario hacia abajo de esta misma página (queremos que haga scroll). Por este motivo cuanto más bajamos en esta página nos iremos encontrando con contenido menos relevante. Esto no significa que los últimos elementos no sean importantes, significa que los que están por encima son de mayor importancia para los usuarios que llegan a nuestra web por primera vez.

Con todos los bocetos encima la mesa tocaba convertirlos en realidad. Para eso el primer paso es elegir el tema de Wordpress que más se adecúa a nuestra misión y a la vez nos ofrece mejores herramientas y opciones. Este proceso es realmente complicado. Existen miles de temas Wordpress y cientos de ellos son de una calidad

altísima. Esto hace que elegir sea realmente complicado y que, aunque siendo estrictos a la hora de elegir candidatos y aunque comparemos sus prestaciones acabemos con al menos una decena de candidatos.

11.4.Implementación de la base de la web

Para realmente elegir entre los candidatos al tema que acabaremos usando la única opción es empezar a implementar y ver si el tema realmente soporta nuestros objetivos.

Este proceso es realmente tedioso: Consiste en intentar reproducir la portada del boceto con cada tema y además ver que tal se verían los tutoriales con cada uno. Esto es necesario porque cada uno ofrece opciones de personalización distintas. Por ejemplo algunos te permiten modificar el color de los elementos, otros ofrecen plantillas de elementos de la portada realmente interesantes, unos tienen un mejor menú que otros, pueden ser más o menos flexibles, las cabeceras de los posts son distintos etc.

También cabe destacar que todas estas limitaciones se pueden evitar modificando el código de los temas y ajustándolos a tus necesidades. Pero no disponía ni del tiempo ni de los conocimientos para llevar a cabo esta tarea que podría acabar convirtiéndose en un pozo de horas que pondría en riesgo el éxito del proyecto. Por este motivo descarté esta opción.

Después de iterar sobre todos los temas candidatos y pasando el proceso descrito, al fin encontré el tema perfecto que se ajustaba a la perfección a la misión del proyecto. Con él, sería capaz de reproducir cada uno de los aspectos que había diseñado en papel y que además tenía un aspecto atractivo y una portada que me gusta mucho. El tema en cuestión se llama Mesmerize, en concreto utilizo la versión gratuita que viene con menos opciones de personalización pero que igualmente es muy completo.

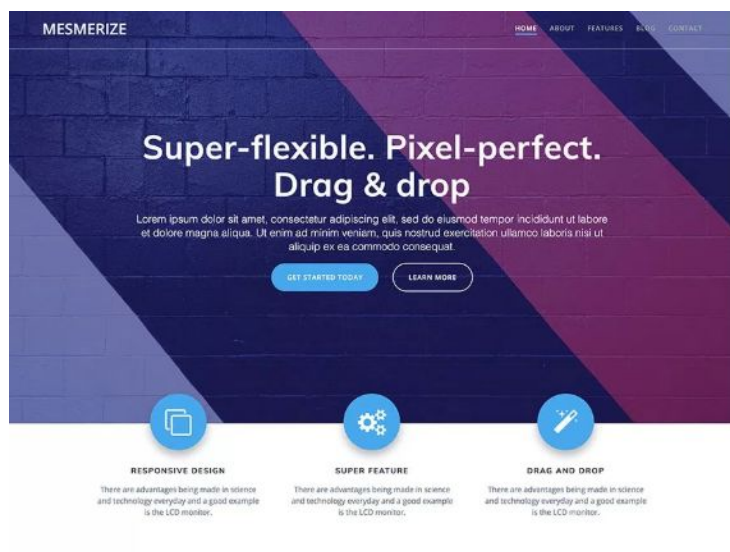


Figura 25: Imágen del tema elegido llamado Mesmerize

Llegados a este punto ya tenía la portada con el formato y apartados actuales, solo que no había imágenes ni el texto final. Lo importante a estas alturas es la implementación de la página, no los detalles.

Cada apartado de la portada es un widget³ incluido en Mesmerize. Estos widgets solo hay que introducirlos donde queramos y configurarlos al gusto con un editor visual super intuitivo y fácil de usar.

Teniendo ya el esqueleto de la portada el siguiente paso fue configurar la barra del menú (la barra que aparece siempre arriba de todo). Esta barra nos permite navegar fácilmente por la página y la vez muestra el icono de Aprender Unity lo que dará identidad de marca y hará que el logo sea asociado con Aprender Unity.

Luego me dediqué a crear el pie de página que habrá en todas las páginas de la plataforma. Aquí me encontré por primera vez con un problema que se repetiría hasta la saciedad a lo largo del proyecto. El problema era que las opciones de Wordpress no me permitían conseguir el aspecto visual que yo deseaba. Entonces, si no hay ninguna opción que nos permita cambiar algo, cómo lo cambiamos? Pues muy fácil, editando el código CSS de la página. Este código define la presentación de la página y establece el diseño visual de los elementos. Así pues modificando este código (que en realidad es más como una hoja de estilo) podemos indicar que elementos queremos modificar, cuales queremos eliminar e incluso cuales queremos crear.

³ Pequeño bloque que realiza una tarea completa. Estos widgets se pueden insertar en los laterales de las páginas y en zonas específicas según el tema o plugins que estemos usando. Como en la portada en nuestro caso.

Aprender CSS fue relativamente fácil dada la gran cantidad de información que hay online y la relativa simpleza de su uso. Así que en unas pocas horas fui capaz de entender cómo modificar el CSS de mi página y de adaptar el pie de página a mis gustos y preferencias.

Teniendo ya el esqueleto de la portada el siguiente paso era hacer realidad el diseño de la página de tutoriales. Este fue uno de los pasos más sencillos ya que Wordpress al fin y al cabo está pensado para hacer blogs, publicar y mostrar entradas fácilmente. Y en el caso de este proyecto los tutoriales son el equivalente a la entrada de un blog. Wordpress también nos facilita el poder establecer diferentes categorías para luego poder acceder y encontrar las diferentes entradas con más facilidad.

Así pues lo primero que hice es añadir dos entradas para testear como se hacía. Y probé a configurar la categoría, añadir una imagen, editar las entradas, darles formato etc. Todo esto está accesible y es muy fácilmente configurable a través del apartado de Entradas que podemos encontrar en el Dashboard de nuestra página Wordpress:



Figura 26: Apartado de Entradas de Wordpress

Teniendo dos posts de ejemplo configuré las opciones de Mesmerize (el tema de la página web) para que su cabecera y el menú lateral fueran de mi agrado. Para la cabecera elegí que se mostrara la imagen del post con un overlay azulado por encima y que se viera el título del post encima:

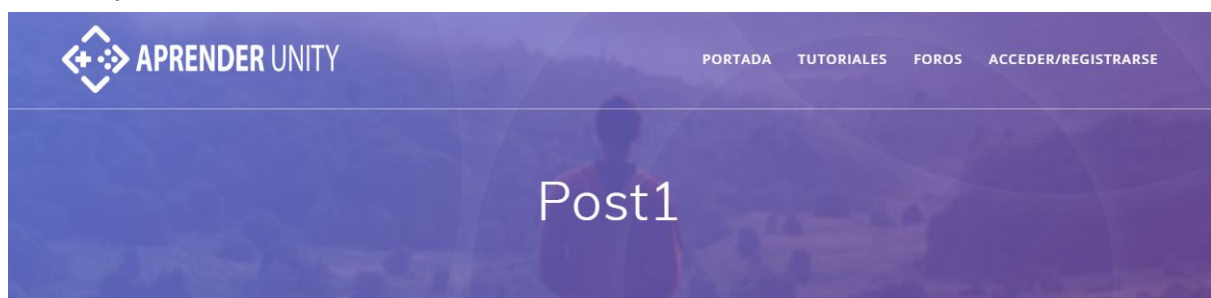


Figura 27: Cabecera de un post de prueba

Por defecto las entradas de una página Wordpress muestran el título de la entrada antes del texto. Así que como ya tenía el título en la cabecera lo eliminé el segundo título del código CSS para que no se mostrara más. Y ya que estaba editando el CSS hice el cuadro del texto de la entrada más ancho para aprovechar más el espacio y no dejar un margen demasiado grande.

En cuanto al menú lateral simplemente lo habilité en el dashboard. Lo configuré de forma que apareciera en las páginas de tutoriales y le añadí los siguientes widgets:

1. Últimos posts: Muestra las últimas entradas añadidas a la página. En nuestro caso muestra los últimos tutoriales.
2. Categorías: Lista de las categorías de las entradas de la página. Si clicamos en una nos lleva a una página con todas las entradas de esa categoría.
3. Suscripción a la lista de correo: Aquí te puedes suscribir a la lista de correo. Tiene un formato distinto (modificado por CSS) que lo hace destacar y llamar la atención del usuario. Añadir este widget dió muchos problemas de los cuales hablaremos en el apartado de problemas a resolver.

El último paso antes de terminar de implementar la base de la web sería actualizar todos los textos e imágenes de la portada, añadir fotos de los diversos proyectos que he realizado este último año y una página para cada uno donde explico algunas de sus características (así muestro la experiencia que tengo programando videojuegos, me vendo como profesor), el apartado de testigos (el cual es creado a partir de las reseñas de un curso de Unity de la asociación VGA que impartí en la UPC) y finalmente configurar el widget de contacto. A través del cual cualquier usuario puede hacerme llegar un mensaje. Ya puestos creé unas cuentas de correo en mi servidor web para así poder recibir correos relacionados con la plataforma, gestionar mejor todo y a la vez hacer de Aprender Unity algo más serio.

11.5.Plugins

Teniendo ya la portada y los tutoriales falta explicar cómo hemos implementado los foros y otras funcionalidades más particulares. Para ello, dado a que necesitábamos funcionalidades que wordpress no incluye hemos tenido que usar plugins⁴ que implementan las funcionalidades que necesitamos. En este apartado se explican todos los plugins que se usan en la web.

A continuación se listan los plugins usados en orden de importancia junto con una pequeña explicación del uso y de la implementación de cada uno:

⁴ Software que contiene un conjunto de funciones que pueden ser añadidas a una página Wordpress. Estas funciones pueden extender funcionalidades existentes o crear nuevas.

Buddypress: Permite convertir cualquier página web Wordpress en una comunidad o en una red social. Permite crear perfiles de usuario, hacer login/logout de la página. Hacer grupos de usuarios, enviar peticiones de amistad, mensajería entre usuarios y todas las funcionalidades del estilo que nos podamos imaginar.

La misión del proyecto era incluir un aspecto de comunidad con un foro y algo que permitiera a los usuarios interactuar entre ellos. Y este es el plugin que lo permite. Aunque tuve que modificar y limitar muchas de sus opciones ya que este plugin soporta muchas más funcionalidades de las que queremos, al final lo único que se usa es el sistema de creación y acceso de los usuarios, la posibilidad de enviar peticiones de amigo, enviar mensajes y el apartado de notificaciones y creación de un perfil asociada a la cuenta de usuario. Esto último opino que es importante porque le da personalidad a los usuarios de la plataforma, hace que las interacciones sean más cercanas.

BbPress: Este plugin extiende las funcionalidades del plugin anterior. Usando las cuentas de usuario y del sistema de notificaciones de BuddyPress nos permite implementar unos foros totalmente personalizables. Estos foros se pueden dividir en categorías, podemos marcar la importancia de los posts, crear diferentes perfiles de usuarios con diferentes permisos, nos permite determinar el contenido de los posts etc...

El gran problema de este plugin es que viene sin formato alguno. Me explico, la mayoría de plugins de Wordpress copian su estilo (su CSS) de la plantilla del plugin que tenemos instalado para así conseguir un aspecto uniforme en toda la web. En cambio este plugin viene con un estilo fijo gris horroroso y con tamaños de los textos demasiado pequeños. Por este motivo tuve que escribir yo mismo toda la plantilla de estilo y aunque el resultado no ha sido excelente está a años luz de lo que venía por defecto.

[Portada](#) › Foros



| Foro | Debates | Publicaciones | Último mensaje |
|---|---------|---------------|--|
| Dudas Unity y programación | 1 | 1 | hace 1 mes  Gerard Belenguer |
| Mostrar trabajo en progreso | 1 | 2 | hace 1 mes  Gerard Belenguer |
| Otros | 1 | 3 | hace 2 semanas, 4 días  Anónimo |

Figura 28: Aspecto final de los foros

El siguiente paso fue añadir a los menús opciones para registrarse, identificarse/login, acceso al perfil (donde podemos interactuar con amigos, cambiar la foto de perfil y de nuestro banner, ver las notificaciones y más) y desconectarse/logout. También modifiqué las barras laterales de las páginas del foro para que tuvieran un widget para identificarse o registrarse. Con estos dos plugins instalados y configurados ya tenía toda la parte social de la plataforma lista pero aún faltaban muchos detalles por pulir.

Theme My login: El sistema de identificación o de login que utiliza Buddypress es el mismo que el que utiliza Wordpress:

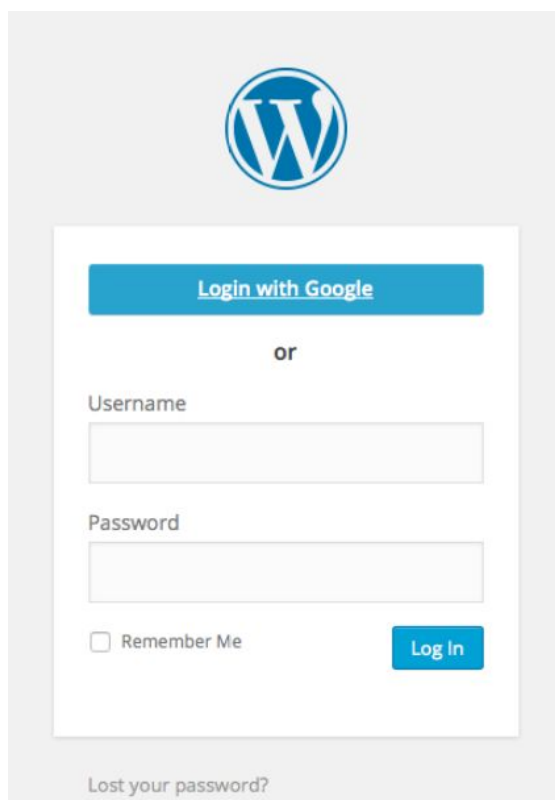


Figura 29: Login por defecto

Este aspecto hace parecer a nuestra web poco profesional e incluso poco seria. Por eso existe este plugin. Lo que hace es convertir esta pantalla al estilo de nuestro tema de Wordpress. Este proceso se podría haber hecho manualmente modificando el código PHP y creando una nueva página pero esta solución era mucho más rápida. Este es el resultado final:

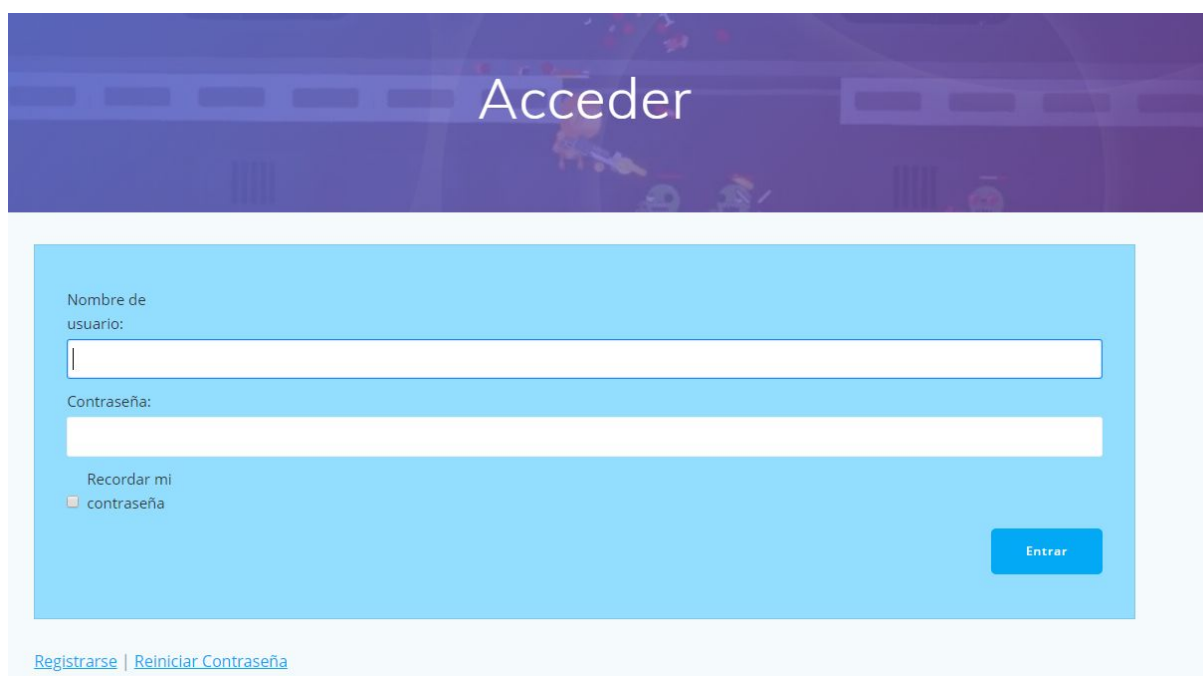


Figura 30: Login final

Yoast SEO: Este plugin es uno de los plugins más usados por los usuarios de Wordpress. Entonces qué hace? Por qué es tan usado?

Yoast se encarga de simplificar el SEO (del cual hablaremos más adelante) de tu página web al máximo y además te ofrece muchas herramientas gratuitas y fáciles de usar para llegar al máximo número de gente tanto a través de navegadores como Google y Bing como a través de redes sociales como Twitter y Facebook. Además también nos permite modificar los títulos, meta descripciones y meta imágenes de cada una de las URLs de nuestra web (lo que se muestra en el navegador y cuando una URL es compartida en redes sociales).

Google Analytics Dashboard: Este plugin es extremadamente importante para el proyecto, dado que este pertenece al campo de los Sistemas de Información. La función de este plugin no es más que la de mostrar los datos de Google Analytics en el Dashboard de nuestra web. Aunque realmente desde la página de Google Analytics contamos con aún más opciones. Pero básicamente esta herramienta sirve para monitorizar todos los aspectos del tráfico de la web, desde la cantidad y la procedencia al comportamiento de los usuarios. Hay mucho que comentar al respecto. Para más información ver el punto 13.1.

Jetpack: Es una versión gratuita y simplificada de lo que ya hace el plugin anterior de Google. Pero además también protege contra algunos ataques maliciosos y te avisa en caso de que la web no sea accesible o esté caída. No es muy importante pero venía instalado por defecto y no está de más tenerlo.

Imagify: Optimiza las imágenes. Reduciendo así el espacio que ocupan y por lo tanto haciendo que la página cargue más rápido. La velocidad de carga es uno de los factores que se tienen en cuenta en el SEO. Así que este plugin mejora la experiencia del usuario y a la vez nos ayuda a posicionarnos mejor en las búsquedas de los navegadores.

Genesis eNews Extended: Nos permite crear un widget a través del cual los usuarios se pueden suscribir a nuestra lista de correo. Esto es un desafío si no contáramos con este plugin ya la gestión de las listas de correo las hacemos a través de MailChimp. Así pues necesitamos una forma de hacer que los datos que recogemos en nuestra web lleguen correctamente a MailChimp. Y este plugin es la forma de hacerlo. Por último me gustaría añadir que el widget viene sin formato y tuve que modificarle el código CSS. Este es el resultado:



Figura 31: Widget retocado de la lista de correo

Mesmerize Companion: Incluye todos los widgets que hemos usado en la portada y un editor muy fácil de usar para ordenarlos y modificarlos.

Elementor: Editor de páginas Wordpress que nos permite añadir widgets y contenido de forma fácil e intuitiva. Muy útil a la hora de crear las páginas con las descripciones de los proyectos que hay en la portada y para crear todos los tutoriales. En los tutoriales nos permite ser completamente flexibles e introducir vídeo, texto, imágenes e incluso un botón de descarga con los contenidos necesarios para seguir el tutorial.

Custom CSS: Nos ofrece la posibilidad de añadir código CSS que sobrescriba al código original. Esto es útil para hacer pequeñas modificaciones de CSS. Este plugin tampoco lo he usado mucho ya que prefiero modificar el código CSS directamente en el servidor ya que es ligeramente más eficiente que hacerlo así.

WP Super Cache: Genera ficheros HTML estáticos a partir de tu página Wordpress. Este fichero se guarda en el servidor y es el fichero que se le sirve al visitante en vez de servirle los pesados scripts PHP de Wordpress. Esto acelera enormemente nuestra página. Lo que aparte de mejorar la experiencia de usuario mejora nuestro posicionamiento dentro de los navegadores ya que es uno de los parámetros a considerar a la hora de posicionar los resultados.

Duplicate Post: Este plugin nos permite duplicar un post o entrada del blog. En nuestro caso esto son los tutoriales. Esto es útil ya que nos permite duplicar posts existentes con toda la estructura hecha. De alguna forma nos sirve para usar plantillas al crear nuevas páginas.

Amr_shortcode_any_widget: Sirve para inyectar widgets en medio del contenido de una página de nuestra web. Si le damos un widget nos retorna un código que si lo introducimos dentro de nuestro código HTML nos inserta dicho widget.

Akismet Anti-Spam: Protege contra diferentes formas de spam y acciones maliciosas. Es uno de los plugins más usados y no está de más tenerlo activado.

11.6.Problemas a resolver

Como era de esperar también hubo problemas a resolver. Estos problemas cambiaron ligeramente la distribución de las horas ligeramente respecto al plan inicial (ver el apartado de Seguimiento para más detalles), pero en general no afectaron al proyecto ya que en la fase de planificación ya había asumido que habrían problemas y dificultades a superar.

Antes de ver cuales son los problemas que tuve a lo largo del proyecto me gustaría mencionar que gran parte de ellos los resolví gracias a una página llamada wpbeginner.com [11] la cual cuenta con todo tipo de soluciones a todo tipo de problemas relacionados con Wordpress. Dicho esto, aquí están los problemas que me encontré en orden de aparición.

Creación y configuración de las cuentas de correo: Este problema realmente es algo tonto. Quería crear cuentas de correo usando mi dominio (aprenderunity.com) y redirigirlos a mi correo gmail de toda la vida. Esto me permitiría leer los correos que recibiera más fácilmente. El problema es que después de hacer la configuración enviaba correos desde el gmail a las direcciones que acababa de crear y el correo se perdía y nunca me llegaba de vuelta. Tras hablar con atención al cliente de SiteGround me dijeron que todo estaba bien configurado y que el problema era que estaba enviando los correos desde el correo al que luego había que redirigir y que eso causaba un bucle.

CSS: Como ya hemos comentado por encima muchos de los elementos de los widgets, plugins y incluso algunas cosas del tema que elegimos o del mismo Wordpress no acababan de encajar con la estética que tenía en mente. Y para arreglarlo tuve que cambiar el CSS.

Realmente aprender CSS es fácil. Pero cuando hay que modificar cosas más complejas como elementos que están anidados dentro de 10 clases y que usan sistemas de layouts algo complejos como un layout por columnas la cosa se complica. Y requiere de ensayo y error, investigación e incluso algo de perspicacia y creatividad para ver cual es la mejor forma de conseguir el resultado deseado.

Integración de MailChimp: Para que el usuario de la web pudiera suscribirse a mi lista de correo de MailChimp necesitaba un widget que enviara la información del usuario a la lista de correo. Para hacerlo existen muchos plugins y opciones pero ninguna parecía encajar a lo que buscaba.

Simplemente quería un widget que pudiera insertar en la barra lateral. Pero todos los plugins que encontraba o no funcionaban, o eran de pago, eran pop-ups o tenían un aspecto que no se ajustaba a lo que quería.

Al final la solución fue usar un plugin poco popular y poco usado. Que aunque requiere una configuración algo más manual realmente funciona, es gratis, no tiene publicidad, es personalizable y en resumen se ajusta perfectamente a lo que buscaba (ver Figura 31).

Gestión de descargas: El objetivo era tener algún tipo de enlace o botón que permitiera descargar archivos desde las páginas de los tutoriales. Esto permitiría por ejemplo compartir los archivos que se necesitan para seguir el tutorial de forma fácil. Para implementarlo lo que al final hago es subir el archivo al servidor y luego crear un botón en el tutorial con un enlace a este archivo.

Problemas con el login a la página: Este problema es seguramente el que más tiempo ha llevado y es el único que realmente ha conseguido hacer superar el tiempo estimado por bastante. Como ya hemos visto en el apartado de plugins el aspecto del login por defecto de Wordpress es poco profesional e incluso puede llegar a asustar a los usuarios ya que parece que estás saliendo de Aprender Unity (Figura 29).

Por este motivo era esencial hacer un sistema de login y registro dentro de la misma web. El plugin Theme My Login resultó ser una solución suficientemente buena (aunque no perfecta). Pero antes de llegar a encontrar y a configurar correctamente este plugin hubieron decenas y decenas de horas perdidas en foros y descargando y testeando plugins con poco éxito.

Muchas cosas fueron mal durante estas pruebas. Algunos plugins no hacían exactamente lo que quería, otros no funcionaban con mi tema, otros funcionaban a la perfección pero eran poco configurables, demasiado complicados para el usuario e incluso hubo un plugin que rompió la web completamente y para arreglarlo tuve que entrar en el servidor y eliminarlo manualmente.

En resumen esta parte ha sido el gran calvario de la implementación de la web y sin duda el mayor problema con el que me he encontrado. Podría haberme rendido antes y quedarme con una de las opciones que medio funcionaban y eran presentables pero después de tanto tiempo invertido y siendo algo tan importante

para el usuario no podía dejarlo a medias. El resultado final puede verse en la Figura 30.

Redireccionamientos: Después de arreglar el problema anterior me dí cuenta que al hacer login y logout se redirigía al usuario a páginas estándar de Wordpress, cosa que no era ideal. Por ello conseguí cambiar el código php para que redirigiera a donde me parecía más indicado en cada caso.

Página en mantenimiento: Wordpress te indica cuando hay actualizaciones para los plugins que tienes instalados. Entonces un día cuando iba a empezar a trabajar en el proyecto, igual que otras veces, vi que tenía varias actualizaciones pendientes. Le di al botón de actualizar todo y a continuación al intentar entrar a la página me encontré el siguiente mensaje:

**Briefly unavailable for scheduled maintenance.
Check back in a minute.**

Figura 32: Mensaje de mantenimiento

Al ver esto me puse decidí esperar un rato. Pero al cabo de media hora seguía igual. Fue entonces cuando algo nervioso me puse a buscar que podría estar ocurriendo y ví que lo que ocurría es que había un fichero .maintenance en la raíz de la instalación de Wordpress en el servidor. Borré dicho fichero y todo volvió a la normalidad.

Forum caching: Cuando implementé de los foros y me puse a testear sus funcionalidades me dí cuenta de que los cambios que hacía no eran inmediatos. Por ejemplo si creaba un nuevo post la página no se refrescaba correctamente. Y si los borraba seguían apareciendo.

Al ver que algo no iba bien me puse a mirar y a tocar todas las opciones de configuración de BbPress (el plugin de los foros) buscando una solución. Al final después de mucha frustración y tiempo perdido me dí cuenta que tenía un plugin instalado llamado WP Super Cache que cachea elementos de nuestra web para acelerar su velocidad. Realmente el plugin funciona muy bien pero en este caso no nos sirve porque al cachear los foros no permite que se actualicen al momento y arruina la experiencia de usuario. La solución fue añadir una excepción a WP Super Cache para que no actúe en las URLs de los foros.

Velocidad de la página: Aunque con el plugin Imagify y con WP Super Cache de los que ya hemos hablado aceleré un poco los tiempos de carga aún no iba del todo rápido y algunos tests de velocidad de páginas [14][15] me sugerían que activara la

compresión gzip. No entraré en detalle de cómo se hace ni en qué consiste exactamente pero este fue el último problema a resolver.

12.Creación de contenidos didácticos

Aprender Unity ha sido creado con el fin de formar a sus usuarios para que aprendan a programar sus propios videojuegos. Cosa que obviamente no sería posible sin crear un buen contenido didáctico.

12.1.Objetivos

Entonces cuál es el objetivo del contenido de la plataforma?

- Base de conocimiento sólida. Los conocimientos se asentarán desde lo más básico.
- Sin rodeos. El objetivo es acortar y digerir todo lo máximo posible,
- Conocimiento útil. Los conocimientos han de ser aplicables en un gran número de proyectos diversos. No se va a enseñar cosas muy concretas aplicables sólo a casos de uso muy específicos.
- Obviamente, como hemos comentado muchas veces ya, el contenido es y será en español.
- Todo el código y todos los assets⁵ usados en los tutoriales estarán disponibles para descargar.
- Para todo tutorial no trivial habrá un vídeo que lo acompañe. La idea es que el tutorial sea un resumen fácilmente accesible y consultable.

12.2.Proceso

Y cuál es el proceso? Cómo conseguimos cumplir estos objetivos? Para garantizar el mejor resultado posible he diseñado una lista de tareas ordenadas necesarias para producir un tutorial. De alguna forma es como el pipeline de creación de tutoriales de Aprender Unity:

1. Pensar el tema del tutorial
2. Destilarlo a los conocimientos más básicos que hay que transmitir
3. Pensar cómo se van a transmitir estos conocimientos. Esto implica pensar qué pequeño proyecto demo se va a crear para el tutorial.
4. Hacer un pequeño esquema dividiendo el proyecto en pasos y indicar qué partes son las más relevantes.
5. Preparar los assets que se van a necesitar para el proyecto.
6. Crear el proyecto y escribir el código. En este paso también se modificaría el esquema del paso 4 en caso que fuera conveniente.

⁵ Cuando hablamos de desarrollo de videojuegos un asset es cualquier archivo que se vaya a usar para crear un videojuego. Esto puede ser una imagen, un sonido, un modelo 3D etc...

7. Crear el tutorial para la web incluyendo un botón de descarga con todos los ficheros.
8. Crear el videotutorial. Editarlo y subirlo a Youtube.
9. Anunciarlo en Twitter.

Este proceso se repetiría para cada uno de los tutoriales, garantizando así una máxima cañidad y que se cumplen todos nuestros objetivos.

13.Gestión del Sistema de Información y presencia online

En este apartado se cubre básicamente todo lo que convierte este proyecto en un TFG de la especialidad de Sistemas de Información. En este apartado se cubren las competencias técnicas del proyecto y se explican todas las herramientas y decisiones que he tomado para poder gestionar esta plataforma. También se explica cómo se va a mantener este sistema de información en un futuro y cómo se gestiona y gestionará la presencia en internet del mismo.

13.1.SEO

La Search engine optimization, normalmente conocida como SEO es el proceso que afecta la visibilidad online de una página web dentro de un navegador. En concreto afecta a los resultados no pagados, normalmente referidos como resultados naturales, orgánicos o merecidos. O sea, es un proceso que afecta el orden en el que se muestran las páginas a los usuarios cuando estos hacen una búsqueda en el navegador.

Los navegadores cuantan con algoritmos para determinar el orden en que se muestran los resultados a las búsquedas dentro del mismo. Este algoritmo varía según el navegador pero excepto algunas excepciones todos funcionan de forma parecida a Google. Además Google es el navegador con más usuarios así que nos interesará modificar adecuadamente nuestra página web de forma que Google considere nuestra página web más importante que las demás y la posicione arriba en la lista de resultados. Esto aumentará la visibilidad de nuestra web y por tanto el flujo de usuarios que acuden a nuestra web de forma orgánica o natural, sin ningún tipo de coste adicional.

Queda claro entonces que dedicarle tiempo al SEO es de vital importancia para el éxito de cualquier página web. Pero antes de ver cómo hemos implementado el SEO de Aprender Unity convendrá saber algunos de los factores que influyen en el

posicionamiento natural de las búsquedas en un navegador (en orden de importancia):

1. Que el contenido responda a la consulta del usuario. Google lo que más busca últimamente es confianza, fiabilidad y efectividad en sus búsquedas. Quiere ser el mejor buscador y por eso esta es su primera prioridad. Para este factor también influye el tiempo que el usuario medio está dentro de la página. Si todos los usuarios salen rápidamente va a tener un posicionamiento malo.
2. Tener keywords acertadas. Estas keywords pueden ser configuradas con el plugin de Yoast SEO y a continuación veremos la estrategia que hemos usado para elegir nuestras keywords.
3. Títulos y metadatos acertados. Es lo que aparece en el buscador. Aquí un ejemplo:

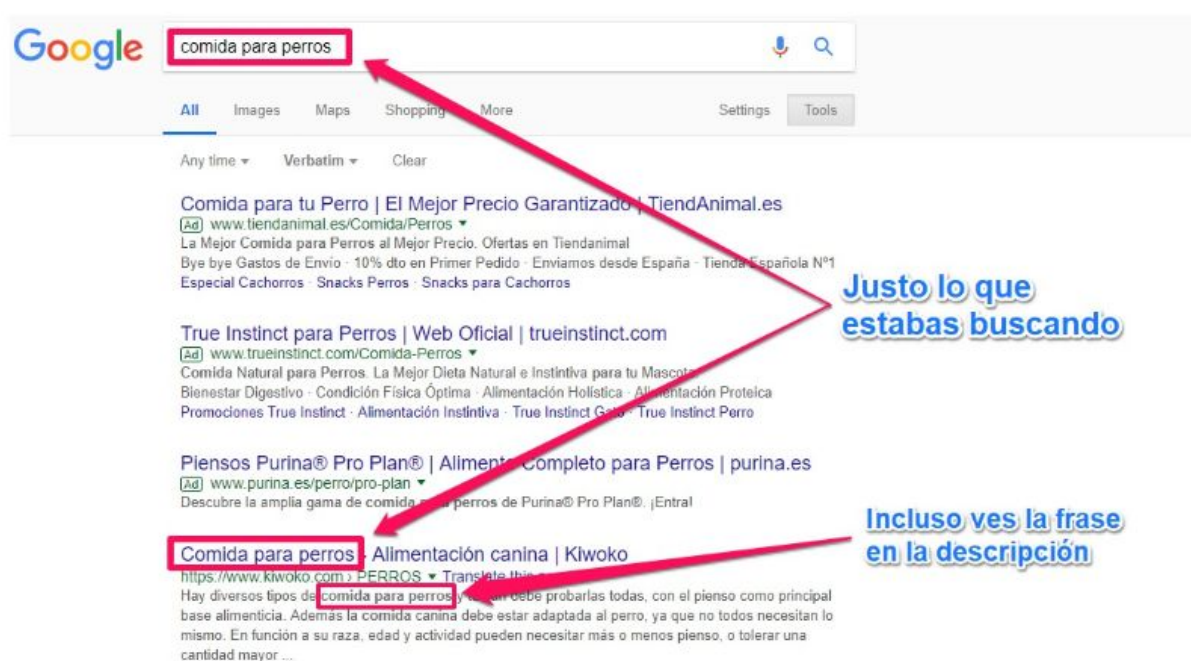


Figura 33: Ejemplo de buen título y meta descripción

4. Optimizado y pensado para teléfono móvil. Por suerte no nos tenemos que preocupar al respecto ya que Wordpress se encarga de esto por nosotros.
5. Velocidad de carga. Este factor es muy importante y como hemos visto en el apartado de plugins y de problemas a resolver hemos hecho varios cambios para acomodar este hecho y asegurar que la página es lo más eficiente posible.

Si nos fijamos los factores 1 y 3 realmente son puro sentido común y realmente no hace falta prestarle especial. Y los factores 4 y 5 ya los tenemos solucionados. Y qué pasa con el factor 2?

El factor 2 necesita de una búsqueda e investigación de las mejores palabras clave que nos vayan a dar mejores resultados. El objetivo es que nuestras palabras clave sea lo que el usuario vaya a introducir en la barra de búsqueda del navegador. Así que lo ideal sería encontrar unas palabras clave muy relacionadas con el contenido y que a la vez fueran las palabras más usadas por tus potenciales usuarios. Otra cosa a tener en cuenta es que tampoco podemos usar palabras muy genéricas aunque estas tengan el mayor número de búsquedas ya que por los factores 1,3,4,5 seguro que hay cientos de webs que saldrán por encima nuestro. Por eso, la mejor estrategia es ser concreto y centrarse en algo específico que a la vez sea probable que tu usuario potencial use en una búsqueda en su navegador.

Si encontrar las palabras clave más acertadas parece complicado es porque realmente lo es. Pero por suerte contamos con algunas herramientas para ayudarnos con la búsqueda. En mi caso he usado Keywords Everywhere [12] y KWFinder [13]. Ambas herramientas son gratis aunque existe mucha oferta de programas similares de pago que igual convendría usar para proyectos con mayor presupuesto.

La primera herramienta Keywords Everywhere es una extensión muy útil de Chrome que nos permite fácilmente encontrar candidatos a buenas palabras clave (o keywords) que más tarde contrastaremos con la segunda herramienta. Keywords Everywhere lo que hace es que cuando hacemos una búsqueda en Google (y en otras páginas como Amazon también pero para este proyecto no nos interesa) nos muestra lo usada y eficaz que es la palabra clave introducida y otras parecidas. Aquí un ejemplo si buscamos aprender unity:

| ★ Keyword | Vol | CPC | Comp |
|-----------------------------------|-------|--------|------|
| ★ programacion en c | 5.400 | \$0.56 | 0.04 |
| ★ unitiescript | 1.300 | \$1.21 | 0 |
| ★ escuelas de videojuegos | 880 | \$1.65 | 0.23 |
| ★ unity scripts free | 260 | \$1.13 | 0.02 |
| ★ udemy c | 210 | \$0.64 | 0.06 |
| ★ unity videojuegos | 140 | \$0.85 | 0.23 |
| ★ unity3d learn tutorials | 110 | \$1.82 | 0.12 |
| ★ curso unity 2d | 110 | \$0.84 | 0.44 |
| ★ unity interactive tutorial | 110 | \$0.38 | 0.04 |
| ★ curso programacion videojuegos | 90 | \$2.48 | 0.91 |
| ★ escuela de videojuegos youtube | 50 | \$0.00 | 0.04 |
| ★ descargar scripts para unity 3d | 40 | \$0.00 | 0.06 |
| ★ curso unity barcelona | 20 | \$2.96 | 0.91 |

Figura 34: Resultados de Keywords Everywhere para “aprender unity”

Como podemos ver “aprender unity” no está en nuestra lista de palabras clave lo que significa que no es algo que la gente busque y por lo tanto no deberíamos usarlo como palabra clave. Pero cómo decidimos qué palabras clave elegir? Qué significan las columnas de Vol, CPC y Comp?

- Vol: Volume. La media de búsquedas por mes a lo largo de 12 meses. Muestra la media de usuarios que hacen esta búsqueda por mes.
- CPC: Cost per click. El coste por click que anunciantes están pagando por un click para esta palabra clave.
- Comp: Competition. Los valores van entre el 0 y el 1. Como más cerca de 1 esté significa que hay más anunciantes.

Un buen ejemplo dentro de la Figura 34 que ilustra que la herramienta funciona es el hecho que “escuelas de videojuegos” tenga un CPC de 1.65\$ y un Comp de 0.23 lo que significa que esta palabra clave es de mucho interés para mucha gente y que están dispuestos a pagar por mostrar anuncios. En este caso los anuncios los paga claramente organizaciones que organizan cursos de videojuegos. Y en efecto si buscamos “escuelas de videojuegos” en Google nos aparece:

Creación de Videojuegos | Estudia máster en Pompeu Fabra | UPF.edu

[\[Ad\] barcelonaschoolofmanagement.upf.edu/master/videojuegos](#) ▼

Programación gráfica, inteligencia artificial, diseño visual, modelado. Infórmate. Prácticas Profesionales. Plan de estudios. 60 créditos.

FP Escuela Videojuegos | Ciclo Formativo Grado Superior | IFP.es

[\[Ad\] www.ifp.es/cfgs/Videojuegos](#) ▼

FP Oficial Online / Presencial. Prácticas Becas & Financiación. Horarios Mañana y Tarde. Títulos FP Oficiales. Opción 100% Online. Cursos: FP Auxiliar de Enfermería, FP Admin. y Finanzas, FP Educación Infantil, FP Desarrollo Web, FP Sistemas Informáticos, FP Comercio y Ventas. FP de Higiene Bucodental · FP Producción Audiovisual · FP Finanzas & Admin. · FP Microinformática

Escuela Videojuegos 3D en BCN | Trabaja de tu pasión | evol3d.com

[\[Ad\] www.evol3d.com/videojuegos/convocatoria](#) ▼

Aprenderás a crear videojuegos. Matricula abierta ¡Solicita información hoy! +de 15 años de experiencia. Convocatoria Abierta. Plazas limitadas. clases reducidas.

Intensivos de Verano · Master Videojuegos · Diseño de videojuegos · Carrera Videojuegos

📍 Plaça Universitat, 5, Barcelona - Open today · 10:00 AM – 1:30 PM ▼

Vols dissenyar videojocs? | Grau en disseny de Videojocs | enti.cat

[\[Ad\] www.enti.cat/Grau/Artist](#) ▼

Representa l'univers d'un joc, modela, textura i anima els personatges a ENTII!

Sortides - Grau Art. · Pla d'Estudis - Grau Art. · Professorat - Grau Art.

Figura 35: Búsqueda de “escuelas de videojuegos” en Google

Y efectivamente vemos que era tal como predecíamos. Lo que podemos sacar de esto es que también tenemos que ir con cuidado con el CPC y Comp ya que si esto es elevado seguro que tendremos gente por encima nuestro en la lista de resultados ya que han pagado para estar allí. Aunque podríamos hacer alguna excepción si por ejemplo pensamos que nuestros usuarios potenciales pueden llegar a nuestra web haciendo esa búsqueda.

Usando Keywords Everywhere y probando con diferentes palabras clave he acabado eligiendo estas palabras clave provisionales:

| Palabra Clave | Vol | CPC (\$) | Comp |
|--------------------------------|------|----------|------|
| unity videojuegos | 140 | 0.85 | 0.23 |
| escuelas de videojuegos | 880 | 1.65 | 0.23 |
| curso programacion videojuegos | 90 | 2.48 | 0.91 |
| juegos hechos con unity | 260 | 0.00 | 0.02 |
| curso unity 2d | 110 | 0.84 | 0.44 |
| programar unity | 880 | 0.85 | 0.17 |
| como crear un juego para pc | 1300 | 0.85 | 0.05 |
| crea tu propio videojuego | 390 | 0.5 | 0.02 |
| crear juegos con unity | 590 | 0.33 | 0.02 |
| aprender unity desde cero | 50 | 1.84 | 0.35 |

Figura 36: Primeros candidatos a palabras clave

Estos primeros resultados que hemos obtenido de forma rápida los vamos a introducir en KWFinder y con suerte nos va a dar más información y mejores sugerencias de palabras clave. Usamos 2 herramientas distintas ya que Keywords Everywhere es una buena herramienta para hacerse una idea de que palabras podríamos usar pero KWFinder es mucho más potente a la hora de decidir ya que da más información y es más fiable. Antes de ver los resultados de este paso vamos a ver un ejemplo de los resultados que nos devuelve KWFinder:

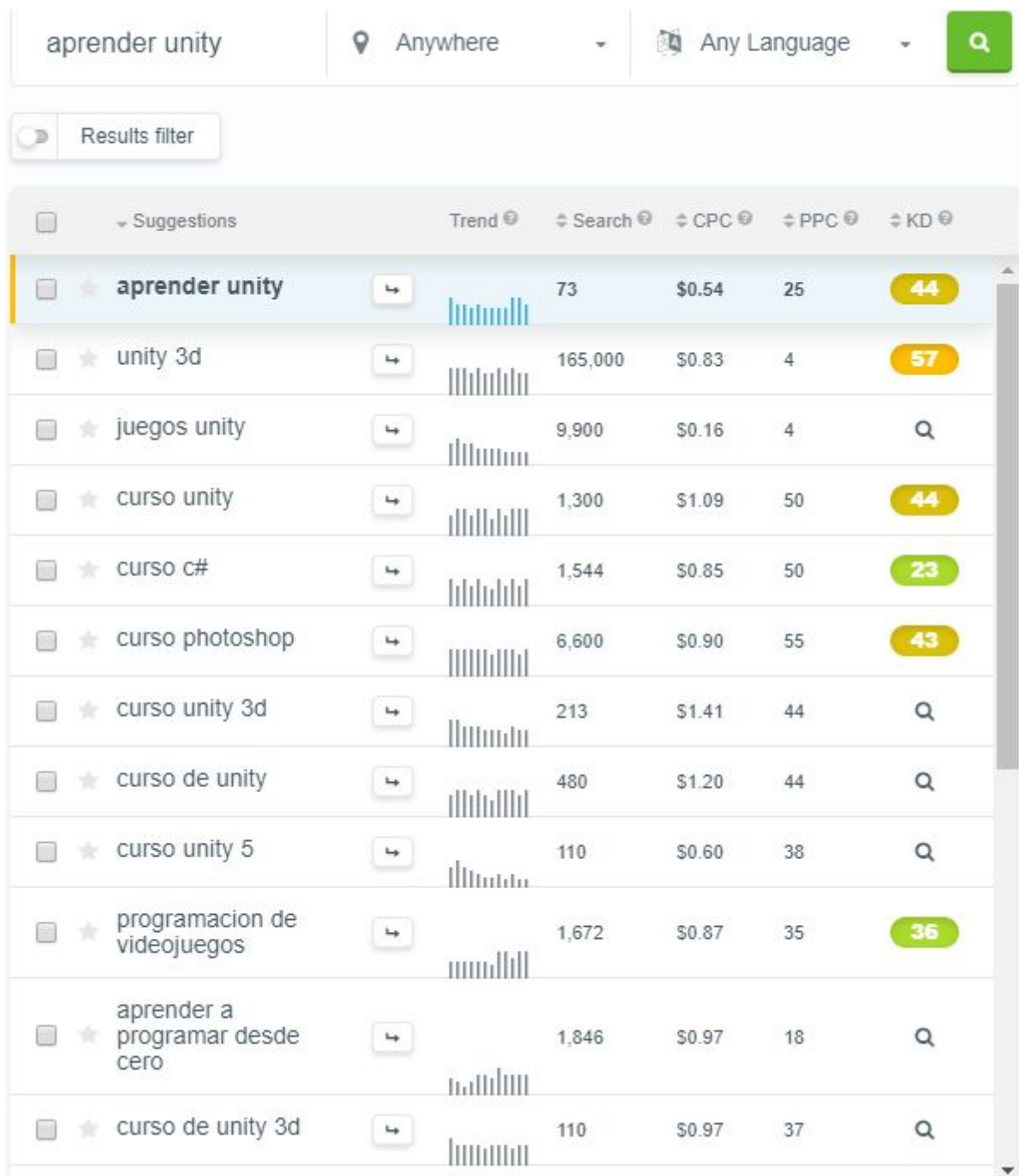


Figura 37: Resultados de "aprender unity" en KWFinder

Como podemos observar los resultados e información son parecidos. Solo que aquí los datos son más precisos y contamos con una nueva métrica muy útil, el KD que significa Keyword Difficulty y nos indica como de mala es la palabra clave y por lo tanto como más cerca de 0 sea este valor mejor.

Estos son los candidatos finales a palabras clave:

| Palabra Clave | Vol | CPC (\$) | Comp | KD |
|-----------------------------|-------|----------|------|----|
| juegos unity | 9900 | 0.16 | 0.04 | 41 |
| curso unity | 1300 | 1.09 | 0.5 | 46 |
| programacion de videojuegos | 1672 | 0.87 | 0.35 | 37 |
| crea tu juego | 18100 | 0.03 | 0.35 | 19 |

Figura 38: Palabras clave elegidas

Estas son las palabras clave elegidas para la plataforma web. Y dentro de estas palabras clave sin ningún tipo de dudas la mejor es “crea tu juego”. Tiene de media 18100 búsquedas mensuales y tiene relativa poca competición comparado con el número de visitas que hay. Además al no haber mucha competencia de contenido similar en español esta va a ser una palabra clave que aportará muchas visitas en un futuro. O al menos, aportará más que el resto de opciones que hemos considerado. Y lo más importante de todo: se ajusta a la perfección a los contenidos de Aprender Unity.

13.2. Estadísticas de la plataforma y toma de decisiones (Google Analytics y Jetpack)

Para poder tomar decisiones en un futuro necesitamos información. Como hemos aprendido a lo largo de la carrera la información no es más que datos tratados, o sea unos datos que cuentan una historia, nos ayudan a llegar a una conclusión a partir de la cual haremos tomaremos mejor nuestras decisiones.

En el caso de Aprender Unity (y realmente cualquier otra web) necesitaremos saber que tal funciona nuestro contenido, decidir qué cambios hacerle a la web y a nuestros contenidos y en resumen tener información que nos ayude a decidir qué pasos tomar en vez de hacerlo todo a ciegas o guiados por la intuición. Hacer caso a la intuición suele ser buena idea pero siempre tenemos que tener información para verificar si estamos en lo cierto.

Existen muchas formas de conseguir esta información pero para este proyecto hemos usado las siguientes dos herramientas:

Google Analytics: Esta es nuestra gran herramienta a la hora de tomar decisiones. Google Analytics es la herramienta más potente (al menos gratuita) que existe a la hora de analizar y monitorizar todo el movimiento que haya en nuestra web e incluso nos permite ver de forma muy exhaustiva el comportamiento y preferencias de nuestros usuarios.

Entonces, sabiendo que tenemos acceso a toda la información que podamos llegar a desear sobre el tráfico de la web y del comportamiento de los usuarios a continuación vamos a explicar la información más importante que extraemos y su utilidad. Hay que considerar que todos los datos que se va a mostrar no se han de considerar como reales ya que aún no ha habido tráfico de usuarios reales. El tráfico que se ve reflejado en las gráficas sale mayormente de la fase de desarrollo, de testeos y de tráfico de prueba realizado o orquestado por mí.

En primer lugar nos interesa saber la procedencia del tráfico. Tanto físicamente como virtualmente. Me explico, por un lado queremos saber desde qué página web proviene nuestro visitante, nos interesa saber si nuestras campañas de Twitter funcionan, si hay mucha gente entrando a través de búsquedas en el navegador, si los suscriptores del canal de Youtube luego pasan por la página web y este tipo de cosas. Por otro lado nos interesa saber dónde viven nuestros visitantes y también su demografía. Si sabemos su procedencia, edad aproximada e incluso el género podremos adaptar mejor nuestro contenido. A continuación adjunto unas imágenes de esta información:

Top Channels

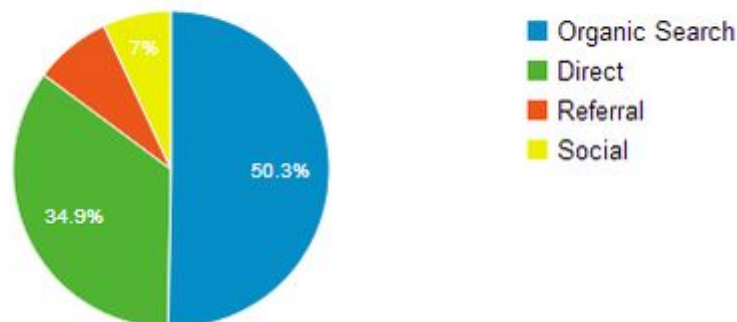


Figura 39: Procedencia del tráfico



Figura 40: Procedencia de los usuarios

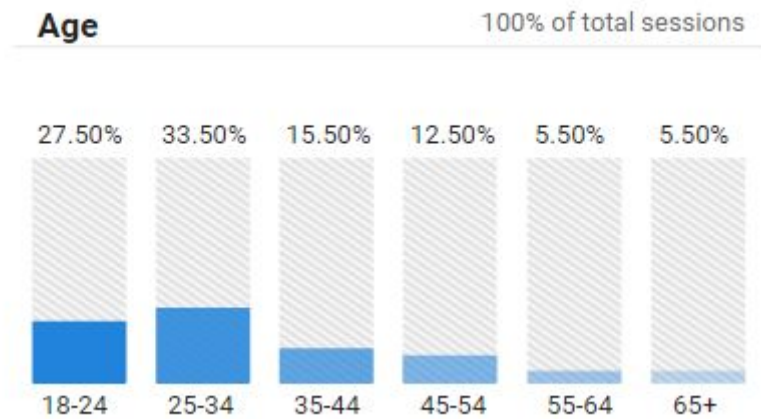


Figura 41: Edad de los usuarios

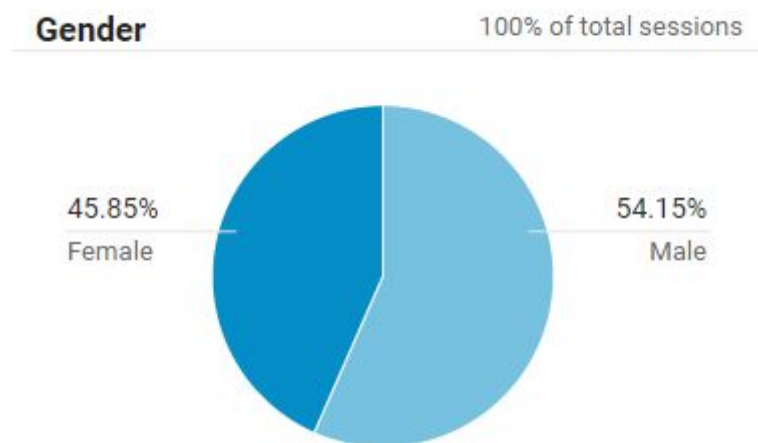


Figura 42: Género de los usuarios

Otro factor muy importante es el número de interacciones por visita y saber la cantidad de gente que retorna a la web (usuarios recurrentes). O sea, la gente que visita la web más de una vez a lo largo del tiempo. Y también nos interesará saber su comportamiento. Toda esta información se puede ver en la siguiente tabla:

| User Type ? | Acquisition | | | Behaviour | | |
|----------------------|-----------------------------------|-----------------------------------|-----------------------------------|---|---------------------------------------|---|
| | Users ? ↓ | New Users ? | Sessions ? | Bounce Rate ? | Pages/Session ? | Avg. Session Duration ? |
| | 16 % of Total: 100.00% (16) | 12 % of Total: 100.00% (12) | 31 % of Total: 100.00% (31) | 48.39% Avg for View: 48.39% (0.00%) | 4.61 Avg for View: 4.61 (0.00%) | 00:03:25 Avg for View: 00:03:25 (0.00%) |
| 1. New Visitor | 12 (70.59%) | 12 (100.00%) | 12 (38.71%) | 58.33% | 2.58 | 00:02:34 |
| 2. Returning Visitor | 5 (29.41%) | 0 (0.00%) | 19 (61.29%) | 42.11% | 5.89 | 00:03:57 |

Figura 43: Nuevos visitantes vs recurrentes

En la tabla se puede apreciar que los visitantes recurrentes pasan más tiempo en la web en sus sesiones y que además interactúan más con la web, hacen más clicks. Lo que queremos en esta tabla es maximizar el porcentaje de usuarios recurrentes y maximizar los tiempos por sesión de ambos tipos de usuarios.

También nos interesará saber en qué página decide el usuario dejar de navegar en nuestra web. Ya que si hay una página donde muchos usuarios deciden salir de nuestra web habrá que arreglarla.

| | | |
|---|--|-------------|
| 1. / | | 22 (70.97%) |
| 2. /blog/ | | 3 (9.68%) |
| 3. /identificacion/?redirect_to=https://aprenderunity.com/ | | 2 (6.45%) |
| 4. /?customize_changeseid=f23aef3d-fd29-4927-a6fa-c4e99309b681&customize_theme=mesmerize&customize_messenger_channel=preview-0 | | 1 (3.23%) |
| 5. /forums/ | | 1 (3.23%) |
| 6. /forums/forum/dudas-unity-y-programacion/ | | 1 (3.23%) |
| 7. /forums/forum/otros/ | | 1 (3.23%) |
| 8. /?customize_changeseid=050b7910-db1c-45e6-b851-f7b10042db80&customize_theme=mesmerize&customize_messenger_channel=preview-0 | | 0 (0.00%) |
| 9. /?customize_changeseid=050b7910-db1c-45e6-b851-f7b10042db80&customize_theme=mesmerize&customize_messenger_channel=preview-1&customize_autosaved=on | | 0 (0.00%) |
| 10. /?customize_changeseid=901e0fed-9a49-491e-a0c0-f9d6850b49b1&customize_theme=mesmerize&customize_messenger_channel=preview-0 | | 0 (0.00%) |

Figura 43: Exit pages de Aprender Unity

Por último también nos interesará saber que tienden a hacer nuestros usuarios cuando entran a nuestra web. Para visualizarlo podemos consultar una tabla de flujo del comportamiento de los usuarios:

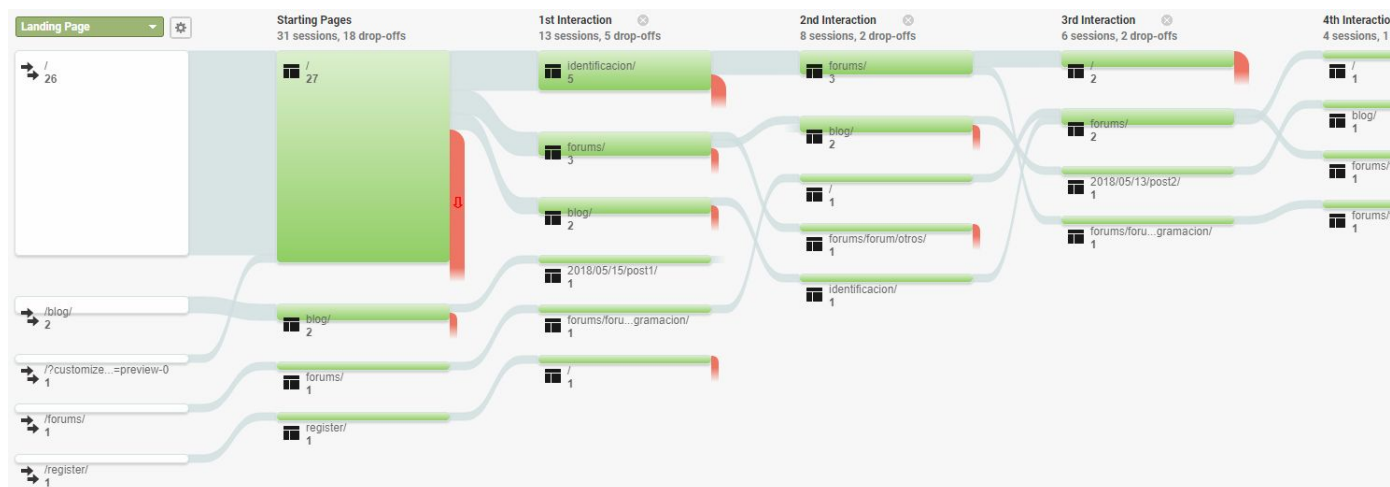


Figura 44: Tabla de flujo de comportamiento

Con toda esta información que sacamos de Google Analytics somos capaces de producir contenido más acertado a nuestra demografía y sobretodo saber decir que está funcionando que no y qué cosas habría que cambiar. Tenemos suficiente

información para tomar decisiones basadas en datos reales de nuestro tráfico y usuarios.

Jetpack: Es un plugin de Wordpress del que ya hemos hablado en el apartado de plugins. Realmente nos aporta poca información y además Google Analytics nos proporciona esta misma información y en mucho más detalle. Pero lo seguimos usando porque nos permite hacernos una idea de cómo está funcionando la web con un simple vistazo en el dashboard de Wordpress. Simplemente entramos en el dashboard clicamos en la pestaña de Jetpack y se nos muestra la información siguiente:

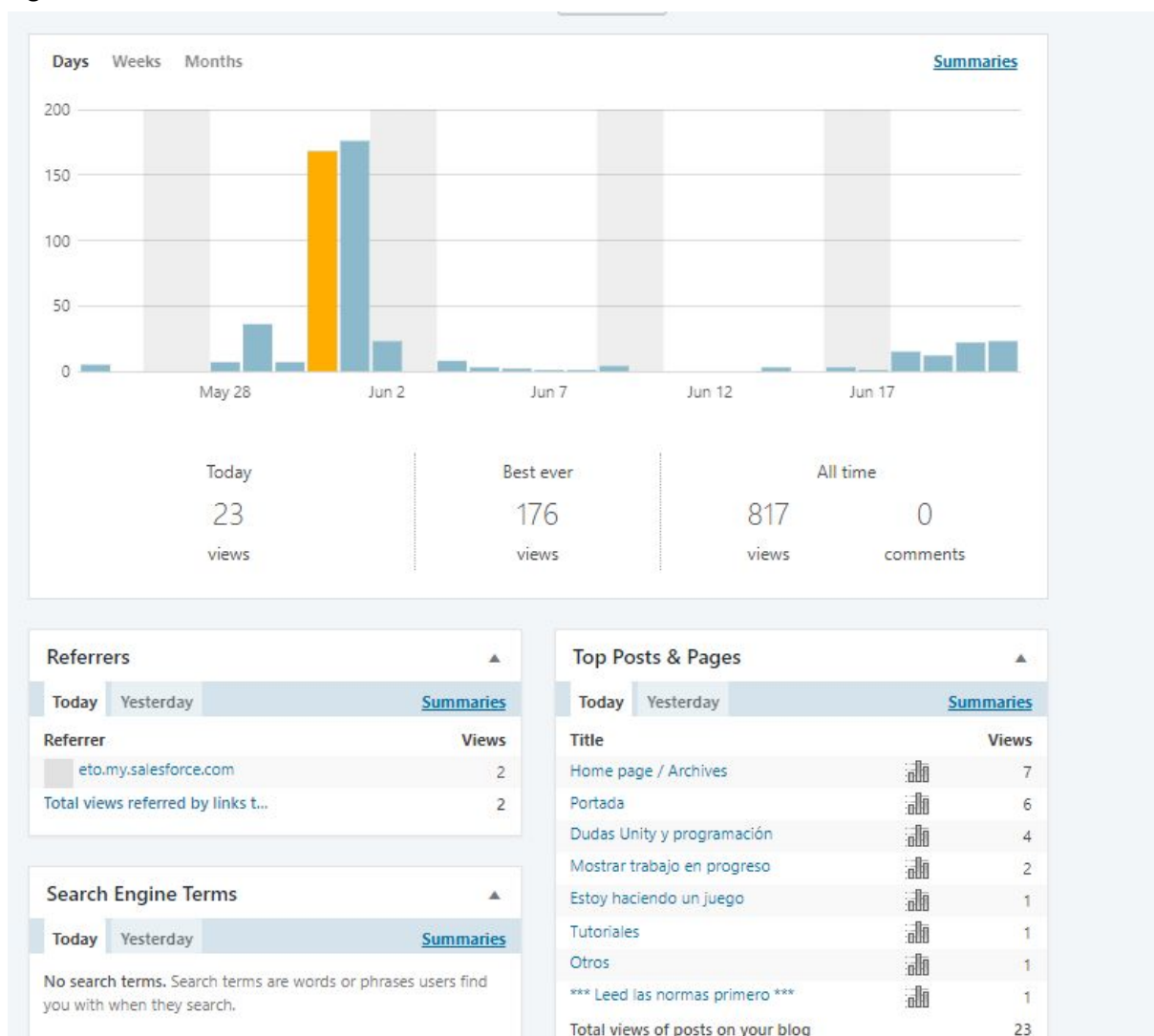


Figura 45: Información de Jetpack

Como se puede apreciar en la Figura 45 Jetpack nos hace un resumen de los elementos más esenciales de nuestra página para que nos podamos hacer una idea clara de cómo de bien o mal está funcionando y de cuáles son las páginas o contenidos más visitados. Además también nos muestra las palabras claves usadas

en un navegador para llegar a la web y enlaces a través de los cuales gente a llegado a la web.

Finalmente con toda esta información tanto de Google Analytics como de Jetpack podemos hacer cosas más interesantes como el A/B testing. Imaginemos que queremos añadir una nueva funcionalidad a la web o cambiar alguna característica. Si tenemos varias opciones en mente podemos saber que versión es mejor? La respuesta es el A/B testing. Este tipo de testing consiste en hacer un experimento controlado con dos (o más) variantes. Luego podríamos implementar estas variantes en diferentes sitios de la web, analizar el comportamiento de los usuarios tratando los datos de navegación en estas variantes y finalmente llegaríamos a una conclusión empírica sobre cuál de las variantes es la mejor (al menos desde el punto de vista de nuestros usuarios).

13.3.Decisiones de monetización

Como hemos visto muchas veces en la especialidad de Sistemas de Información al fin y al cabo todo proyecto informático tiene gente detrás. Gente que suele invertir mucho tiempo en estos proyectos. Esta gente también tiene que comer y vivir su vida y para ello necesita dinero. Lo que quiero decir con esto es que, excepto contadas excepciones, ha de haber dinero detrás de todos los proyectos informáticos ya que todo tiene un coste, tanto de tiempo como de dinero. Y nadie va a invertir mucho tiempo y dinero a cambio de nada. Tener un objetivo noble y una causa por la que trabajar está muy bien, pero sin dinero uno no puede vivir y ni siquiera mantener el proyecto vivo.

Por eso cómo se va a monetizar Aprender Unity? Ya que de momento lo único que he hecho es invertir algo de dinero y mucho tiempo. Pues la respuesta es que de momento no se va a monetizar, va a ser completamente gratuito.

Esta afirmación contradice completamente lo dicho en el primer párrafo, pero hay un motivo para ello: ahora mismo no hay forma de monetizar correctamente la plataforma. No hay volumen de usuarios suficientes para hacer nada así pues cualquiera de las miles de estrategias de monetización nos aportaría una cantidad irrisoria de dinero. Y en vez de ganar este dinero a cambio de una peor experiencia para el usuario prefiero empezar ofreciendo todo de forma 100% gratuita y abierta.

Esto va a permitir, hipotéticamente claro, que la plataforma crezca más rápido y que en un futuro, cuando hayan muchos más usuarios se pueda monetizar el contenido ya sea con anuncios, contenido exclusivo, cursos de pago, patronaje etc...

13.4.Gestión de las listas de correo

Como ya se ha comentado en otros apartados los usuarios y visitantes de la página tendrán la oportunidad de inscribirse a la lista de correo, muchas veces también llamada newsletter. Tener una lista de correo es una de las mejores herramientas de marketing que hay. Pero por qué? Qué nos aporta?:

- Conciencia: Es una oportunidad de hacer que el cliente sea más consciente de los objetivos y de la perspectiva de la empresa, o de la plataforma web en nuestro caso.
- Promoción: Puedes informar de nuevos productos, servicios e incluso ofrecer por ejemplo códigos promocionales de descuento. Esto sería útil en nuestro caso para informar a los usuarios de nuevo contenido en la página o para ofrecer un código de descuento a los seguidores de la lista de correo si alguna vez se crea un curso de pago exclusivo.
- Contacto: Escribir regularmente a los usuarios ayuda a formar una relación que a la vez fideliza al usuario.
- Interacción: También podemos usar la lista de correo para interactuar con los usuarios y pedirles opinión o feedback. Por ejemplo en nuestro caso podríamos hacer una encuesta exclusiva para los lectores de la lista de correo para decidir el tema del próximo tutorial. Esto ayudaría a que el usuario se sintiera más integrado e relevante dentro de la plataforma.

Sabiendo cuales son las ventajas de tener una lista de correo vamos a ver las dos formas que hay en aprender Unity de inscribirse a la lista de correo que manejaremos a través de MailChimp:

- A través del plugin Genesis eNews Extended: El plugin nos ofrece la posibilidad de añadir un widget en la barra lateral de las páginas y luego envía la información del usuario a MailChimp. Toda la explicación puede verse en el apartado de Plugins y el resultado lo podemos ver en la Figura 31.
- A través de un pop-up de MailChimp: Desde la página de MailChimp podemos crear distintos formularios de registro a nuestras listas de correo. Una de las opciones que se nos ofrece es crear un formulario en forma de pop-up. Así pues diseñamos el formulario dentro de un editor. Una vez tenemos el resultado deseado podemos copiar el código del pop-up y usarlo en nuestra página Wordpress en la parte que creamos más conveniente. En el caso de Aprender Unity para hacer que este pop-up no fuera demasiado intrusivo lo configuré de forma que apareciera en la portada cuando el usuario ha llegado a la mitad del contenido de la página. Este es el resultado:

A pop-up form for email subscription. The top section has a blue background with a red mailbox icon and several white envelopes. Below this, the text reads: "Subscríbete a nuestra lista de correo para recibir una lista exclusiva de los 10 consejos sobre desarrollo de videojuegos". There are two input fields: "Email" and "Nombre". The form is enclosed in a grey border with a close button (X) in the top right corner and a scrollbar on the right side.

Subscríbete a nuestra lista de correo para recibir una lista exclusiva de los 10 consejos sobre desarrollo de videojuegos

Email

Nombre

Figura 46: Formulario pop-up de la lista de correo

Teniendo todo esto configurado ya existe la posibilidad de obtener nuevos suscriptores para nuestra lista de correo. Entonces, llegados a este punto, conviene saber que vamos a hacer con estos suscriptores y como vamos a gestionar la lista.

El primer paso sería enviar un correo de forma automática inmediatamente al usuario que se suscriba. Esto es útil ya que el usuario así ve que ya forma parte de la lista de correo de Aprender Unity y por otro lado porque como podemos ver en la Figura 46 en el formulario pop-up prometemos que si alguien se suscribe le enviaremos 10 consejos sobre cómo empezar a programar videojuegos. Esto lo haríamos creando una campaña de MailChimp y configurándola de forma que esta se activara cuando alguien se suscribe a la lista. Finalmente solo haría falta crear el correo de bienvenida que ha acabado teniendo esta forma:

APRENDER UNITY

CREA TUS PROPIOS VIDEOJUEGOS



Bienvenido a Aprender Unity

Me alegra que hayas decidido emprender tu aventura de creación de videojuegos con nosotros. Antes que te des cuenta serás todo un experto, programar videojuegos no tiene por qué ser difícil!

Pero antes de ponernos manos a la obra me encantaría darte mis mejores consejos sobre desarrollo de videojuegos:

1. **Empezar con proyectos pequeños.** Ser ambicioso en los primeros proyectos es el peor fallo que puedes cometer al desarrollar tus primeros juegos. En cambio crea un juego pequeño, simple y pulido. Aprende el proceso, disfruta y luego enséñaselo a tus amigos!
2. **Acaba tu primer juego.** Seguramente será tu peor creación pero es importante que la acabes y veas que eres capaz de crear y acabar un juego.
3. **El detalle es lo que diferencia a los grandes juegos del resto.** Calidad antes de cantidad!
4. **No sabes qué crear?** Cópiate del resto, no hay nada de malo en ello. Inspírate en otros juegos y aprende de ellos.
5. **Estás atascado con algún problema?** Pide ayuda. A los programadores

Figura 47: Parte del correo de bienvenida a la lista de correo

Además de este correo de bienvenida se enviarán correos periódicos que exploten las posibilidades que ofrece una lista de correo y que hemos comentado al principio del apartado. Pero básicamente consistirá en información de nuevos tutoriales, contenidos y actualizaciones de la plataforma, pedir feedback a los usuarios y finalmente informar de promociones.

13.5. Generación de leads y embudo de conversión

Para poder hablar de la generación de leads de la plataforma primero tendremos que saber que es un lead. En marketing, cuando hablamos de leads estamos hablando de un registro de un individuo que muestra interés en la adquisición de un bien o la contratación de un servicio. O sea, es cuando un usuario tiene lo que algunos llaman actitud “compradora” y conseguimos que se registre a nuestro servicio.

Sabiendo que son los leads podemos plantearnos cuales son las principales formas de obtención de leads en el marketing online. Las vías más comunes son:

- Email marketing
- Banners y anuncios
- Marketing en buscadores
- Acciones en redes sociales
- Anuncios en formato vídeo

Aprender Unity solo cuenta con Email Marketing con la lista de correo gestionada en MailChimp que hemos detallado en el anterior apartado. Aunque en un futuro no se descarta usar anuncios ya sea en forma de banners, en navegadores o redes sociales.

Cada vez los usuarios son bombardeados con más publicidad y calls to action cuando navegan online. Un call to action o CTA en marketing se refiere a cualquier dispositivo diseñado para provocar una respuesta inmediata en el usuario como ahora comprar un producto, suscribirse a un servicio o inscribirse a una lista de correo. Por eso los usuarios son cada vez más inmunes a estas call to action como ahora nuestro pop-up de inscripción a la lista de correo.

Por este motivo necesitamos un incentivo para hacer más efectivo este pop-up y conseguir que los usuarios introduzcan su correo y así se conviertan en leads. Estos incentivos tienen un nombre en marketing, se llaman lead magnets (imán de leads). Un lead magnet es un incentivo que se ofrece a potenciales leads a cambio de que estos realicen una acción como introducir su correo electrónico que es precisamente nuestro caso. Como se puede ver en la figura 46 nuestro lead magnet son 10 consejos para desarrollar videojuegos. Igual no es el mejor lead magnet del mundo pero de momento no tengo nada más a ofrecer.

Entonces ahora que tenemos un método de generación de leads con un call to action y un lead magnet nos gustaría poder medir cómo de eficaz es este método. Queremos saber cuántos de los usuarios que ven el pop-up acaban inscribiéndose en la lista de correo. Para ello usamos Google Analytics. Este sería el aspecto que tendría una vez está todo configurado:



Figura 48: Embudo de conversión lista de correo

Como se puede ver en el embudo de la Figura 48 de momento solo hemos tenido 5 visitas que han visto el pop-up y de los cuales ninguna de ellas ha acabado inscribiéndose a la lista de correo. Aunque en el punto en el que se escribe esto la página lleva activa pocos días y a pensar tiene contenido así que necesitaremos más contenido y más información para poder llegar a alguna conclusión.

13.6.Gestión de las redes sociales

Todos sabemos como de importantes son a día de hoy las redes sociales. En especial si queremos tener una presencia importante en internet. Por este motivo será esencial hacer una buena gestión de las redes sociales de Aprender Unity con tal de mejorar la relación con los usuarios y con tal de conseguir usuarios nuevos. De momento las redes sociales que se usan son Youtube y Twitter pero en un futuro, cuando la web tenga más usuarios no se descarta crear un Instagram que es la gran red social de moda y sería una gran herramienta para compartir avances de los tutoriales, contenidos, promociones y actualizaciones venideras.

Pero de momento vamos a centrarnos en lo que hay:

Youtube: Es la plataforma donde se van a colgar todos los videotutoriales. La gestión que hay que hacer en esta plataforma es mínima ya que prácticamente lo único que hay que hacer es subir contenido (videotutoriales) y asegurarse de que ponemos buenas etiquetas a nuestros vídeos para conseguir más visitas (de forma muy parecida a lo visto en el apartado de SEO, de hecho podemos usar Keywords

Everywhere para ello). Aunque nos interesara ser muy activos en los comentarios de los vídeos con tal de fidelizar a los suscriptores del canal.

Por otro lado recordar que el objetivo de este canal de Youtube es tener una plataforma donde subir vídeo y a la vez tener un alcance potencial elevado ya que la visibilidad que te da Youtube puede ser enorme. Estos vídeos también se incluirán en los posts de los tutoriales para facilitar el entendimiento de los vídeos y al final de cada vídeo se invitará al espectador a visitar Aprender Unity y a participar en el foro.

Esto provocará que haya gente que solo use la plataforma web, gente que solo utilice Youtube y gente que use ambas. Pero en todo caso todas las opciones nos sirven, el usuario que solo esté en Youtube solo le interesará el conocimiento y no el aspecto de la comunidad. Lo esencial es crecer y conseguir una base de seguidores y que a la vez tengan la oportunidad de formar parte de una comunidad de gente parecida a ellos. Que luego no todo el mundo haga uso de el aspecto de la comunidad no es importante, lo importante es que esté la opción y que haya una base de usuarios que sí haga uso de la opción.

También mencionar que ha habido un trabajo artístico detrás del arte del canal. Este es el aspecto actual del canal de Youtube:



Figura 49: Banner del canal de Youtube

Twitter: En esta red social habrá que ser mucho más activo y hábil para conseguir seguidores. Cuando no tienes visibilidad alguna es difícil crecer en Twitter. Para ello consideré crear un bot que buscará contenido de creación de videojuegos en español y hiciera Retweets y follows según unos patrones que definiera. Pero al final decidí que no era para nada buena idea ya que hacer esto daría bastante trabajo y al final el resultado sería malo porque acabaríamos teniendo un contenido y forma de interactuar muy artificial que olería mucho a “spam”.

Por este motivo he optado por una gestión manual utilizando una herramienta oficial de Twitter que nos facilita enormemente lo que queremos conseguir. Esta herramienta es TweetDeck. Desde esta página se nos permite programar Tweets, ver

las interacciones que hemos tenido con otros usuarios, leer y contestar mensajes, gestionar nuestros seguidores y todo lo que podemos hacer desde Twitter pero de forma mucho más rápida y eficiente. Pero la funcionalidad que más nos interesa que aún no comentado es la seguir todo el contenido que se produzca y se haya producido bajo una búsqueda concreta, ya sea una palabra o palabras o un hashtag.

Hacer este seguimiento nos permite por ejemplo seguir todos los elementos que pueden ser relevantes para Aprender Unity y donde pueda haber posibles usuarios futuros con los que interactuar con Retweets, Likes y Follows. Un ejemplo podría ser seguir para luego poder interactuar a: “#unity”, “desarrollo videojuegos”, “programación videojuegos” y todo lo que se nos ocurra! Aquí una captura de pantalla para mostrar lo fácil y práctico que es:



Figura 50: Seguir búsquedas con TwitterDeck

Para que esto nos fuera realmente útil tendríamos que acabar de encontrar búsquedas mejores y configurar mejor las opciones de filtrado pero esta es la idea. Además es muy útil ya que podemos tener TweetDeck abierto en un segundo monitor y se irá actualizando y mostrando nuevos tweets de los temas que sigamos automáticamente.

El perfil de Twitter también necesita su propio banner así que reutilizando algunos de los elementos del banner de Youtube creé otro para Twitter. Este es el aspecto de la cuenta cuando estaba recién creada:



Figura 51: Aspecto de la cuenta de Twitter

14. Conclusiones

14.1. Cumplimiento de los objetivos

Una vez llegados al fin del proyecto podemos afirmar que los objetivos que nos habíamos propuesto (y que podemos ver en el apartado 4) han sido cumplidos y que a lo largo de algo más de dos meses de implementación he sido capaz de materializar en la plataforma web todo aquello que fue planificado durante la fase de GEP.

El Scrum modificado que planteé en la fase inicial también ha resultado ser extremadamente eficaz y útil a la hora de acabar el proyecto a tiempo y mantener el orden.

También por suerte han habido pocos incidentes a lo largo del proyecto. Y los que han habido se han compensado con tareas que he podido acabar antes de lo previsto así que la planificación inicial también ha resultado ser bastante acertada.

Aunque como hemos visto en el apartado de seguimiento hubo un ligero rebalance de las prioridades del proyecto con tal de potenciar más los aspectos de sistema de información de la plataforma a cambio de crear menos contenido didáctico. Opino que esta decisión ha sido muy acertada ya que ahora que cuento con más herramientas seré capaz de tomar mejores decisiones y hacer una mejor gestión de la plataforma. Y a su vez ahora tengo todo el tiempo del mundo para crear contenido para la plataforma ya que el proyecto se ha terminado.

14.2. Conocimientos aplicados y adquiridos

Por suerte este proyecto se adecuaba a muchas cosas vistas a lo largo de diversas asignaturas de la especialidad de Sistemas de Información. Para empezar gracias a Diseño de Sistemas de Información (DSI) tenía un conocimiento amplio sobre las diferentes metodologías ágiles que existen y esto me ha permitido evaluar todas las opciones y elegir lo que más me convenía.

Luego, gracias a Ingeniería de Requisitos (ER) he podido plantear los requisitos del proyecto, los casos de uso, historias de usuario y demás características que conviene tener en mente desde el principio. Otra asignatura muy importante ha sido Proyectos de Sistemas de Información (PSI), la cual me ha ayudado mucho a tener una mejor visión y perspectiva sobre en qué consiste realmente un proyecto y como se debe enfocar su confección y desarrollo. Y además me ha ayudado a elegir correctamente qué herramientas tecnológicas iba a usar. Dada la limitación de tiempo, considerando mis aptitudes, necesidades y la visión del producto. Y ahora que ya está todo hecho dudo que exista una herramienta más conveniente que Wordpress para llevar a cabo este proyecto.

Finalmente también me gustaría mencionar Negocio Electrónico (NE) que me ayudó a tener una mejor percepción sobre el mundo de los Sistemas de Información y sobretodo porque el trabajo que hicimos con Prestashop me ha ayudado a la hora de usar Wordpress ya que se parecen bastante. En esta asignatura el profesor también nos dio unas nociones básicas que he podido aplicar sobre cómo mejorar la relación con el cliente para así poder fidelizarlo.

Por otro lado hay muchos conocimientos que he tenido que ampliar a lo largo del proyecto como ahora el conocimiento que tenía sobre marketing en internet y presencia online. Pero sobretodo lo que ha sido más nuevo para mí ha sido crear una página web, aprender a programar en PHP, a editar el código CSS de una web y a administrar un servidor.

En resumen el proyecto ha sido extremadamente útil para poder poner en práctica, consolidar y ampliar todos los conocimientos que he obtenido a lo largo de la especialidad y a la vez ganar algo de experiencia en un proyecto real. Por otra parte estoy muy contento de mis nuevos conocimientos de desarrollo web que he obtenido.

14.3.El futuro de Aprender Unity

Con todas las limitaciones que hemos hablado y la corta duración de este proyecto Aprender Unity aún está en su infancia. De hecho justo acaba de nacer. Cuál es el futuro de Aprender Unity entonces?

Ahora que este proyecto ya se da por acabado y que voy a tener menos presión y más libertad voy a dedicarme a crear más contenido, mucho más contenido y sobretodo mejorar la calidad del contenido y de la producción constantemente. De mientras voy a moverlo por las redes y a intentar atraer más gente de las formas ya descritas.

Así pues, con suerte voy a conseguir tener una base de usuarios y si todo va bien en un futuro no descarto seguir trabajando en la web, probar a usar anuncios de pago para expandir la plataforma aún más y monetizar algunos contenidos.

Pero todo esto ya se verá, de momento es cuestión de trabajar día a día con ilusión y si no sale nada provechoso de este proyecto me quedaré con la experiencia, con los los buenos recuerdos y con los conocimientos aprendidos.

15. Referencias

- [1] Nikola Muckajev. (2017). A New Realm Of Possibility: The Democratization of Game Development. Retrieved April 7, 2018, from <https://www.theversed.com/65695/the-democratization-of-game-development/#.Q1523q0993e1528c6>
- [2] Unity Technologies. [online]. Unity - Manual: Creación de Juego. Retrieved March 3, 2018, from <https://docs.unity3d.com/es/current/Manual/CreatingGameplay.html>
- [3] UnitySpain. [online]. unityspain. Retrieved March 3, 2018, from <http://www.unityspain.com/>
- [4] AgileManifesto. [online]. Manifesto for Agile Software Development. Retrieved March 3, 2018, from <http://agilemanifesto.org/>
- [5] Trello. [online]. * Tableros | Trello. Retrieved March 3, 2018, from <https://trello.com/>
- [6] Extreme Outer Vision. (n.d.). Power Supply Calculator - PSU Calculator | OuterVision. Retrieved March 19, 2018, from <https://outervision.com/power-supply-calculator>
- [7] How To Make A WordPress Website 2018 - YouTube. (n.d.). Retrieved March 15, 2018, from https://www.youtube.com/watch?v=kDNTAYLb_vQ
- [8] How To Make a WordPress Website - 2018 - In 24 Easy Steps - YouTube. (n.d.). Retrieved March 15, 2018, from <https://www.youtube.com/watch?v=2cbvZf1jIJM>
- [9] How To Create A Wordpress Website 2018 | Divi Theme - YouTube. (n.d.). Retrieved March 15, 2018, from https://www.youtube.com/watch?v=1l5OnKj_-5Q
- [10] WPBeginner - Beginner's Guide for WordPress. (n.d.). Retrieved March 15, 2018, from <http://www.wpbeginner.com/>
- [11] Top Ten Rules of Web Design. (n.d.). Retrieved March 18, 2018, from https://sharpened.com/web_design_rules
- [12] FREE browser addon for search volume & cpc of all your favorite keyword tools. (n.d.). Retrieved June 1, 2018, from <https://keywordseverywhere.com/>
- [13] KWFinder - Keyword research and analysis tool. (n.d.). Retrieved June 1, 2018, from <https://kwfinder.com/>

[14] PageSpeed Insights. (n.d.). Retrieved June 15, 2018, from <https://developers.google.com/speed/pagespeed/insights/>

[15] GTmetrix | Website Speed and Performance Optimization. (n.d.). Retrieved June 15, 2018, from <https://gtmetrix.com/>

15.1.Lista de figuras

*No todas las figuras han sido sacadas de un sitio web. Por eso en esta lista no figuran todas las figuras que aparecen en esta memoria

[Figura 1] YouTube. [online]. Brackeys - YouTube - YouTube. Retrieved March 3, 2018, from https://www.youtube.com/channel/UCYbK_tjZ2OrIZFBvU6CCMiA

[Figura 2] YouTube. [online]. Sebastian Lague - YouTube - YouTube. Retrieved March 3, 2018, from <https://www.youtube.com/channel/UCmtYQOKKmrMVaKuRXz02jbQ>

[Figura 3] YouTube. [online]. quill18creates - YouTube - YouTube. Retrieved March 3, 2018, from <https://www.youtube.com/user/quill18creates>

[Figura 4] YouTube. [online]. Hagamos Videojuegos - YouTube - YouTube. Retrieved March 3, 2018, from <https://www.youtube.com/user/juande>

[Figura 5] YouTube. [online]. Mamboro - YouTube - YouTube. Retrieved March 3, 2018, from <https://www.youtube.com/channel/UC7HpST1Viyqy8B12KVVcCng>

[Figura 6] YouTube. [online]. La Cueva De L_Draven - YouTube - YouTube. Retrieved March 3, 2018, from https://www.youtube.com/channel/UC49ACIHmz1yvHqdDE4Y_foQ

[Figura 7] Udemy. [online]. Master en Programación de Videojuegos con Unity y C# 2017 | Udemy. Retrieved March 3, 2018, from <https://www.udemy.com/master-programacion-de-videojuegos-con-unity-5-y-csharp/>

[Figura 8] Udemy. [online]. Learn to Code by Making Games - Complete C# Unity Developer | Udemy. Retrieved March 3, 2018, from <https://www.udemy.com/unitycourse/>

[Figura 9] Coursera. [online]. Cursos de Computer Science | Coursera. Retrieved March 3, 2018, from <https://www.coursera.org/courses?languages=es&query=unity>

[Figura 10] Domestika. [online]. Cursos online de Unity 5 | Domestika. Retrieved March 3, 2018, from <https://www.domestika.org/es/courses/software/42-unity-5>

[Figura 21] Hosting WordPress -Tarifas Mensuales. (n.d.). Retrieved March 28, 2018, from <https://www.siteground.es/hosting-wordpress.htm>

[Figura 22] Top Ten Rules of Web Design. (n.d.). Retrieved June 16, 2018, from https://sharpened.com/web_design_rules